



РЕПУБЛИКА БЪЛГАРИЯ
Министър на образованието, младежта и науката

ЗАПОВЕД

№ РД 09 – 5531/07.05.2013 г.

На основание чл. 36, ал. 2 от Закона за професионалното образование и обучение, във връзка с чл. 42 от Наредба № 3 от 15.04.2003 г. за системата на оценяване, при спазване изискванията на чл. 66, ал. 1 и ал. 2 от Административнопроцесуалния кодекс и във връзка с организирането и провеждането на държавните изпити за придобиване степен на професионална квалификация за професията

УТВЪРЖДАВАМ

Национална изпитна програма за провеждане на държавни изпити за придобиване на трета степен на професионална квалификация за професия код 481020 “Системен програмист”, специалност код 4810201 “Системно програмиране” от професионално направление код 481 “Компютърни науки” от Списъка на професиите за професионално образование и обучение по чл. 6 от Закона за професионалното образование и обучение.

Контрол по изпълнението на заповедта възлагам на Мариана Банчева – заместник-министър.

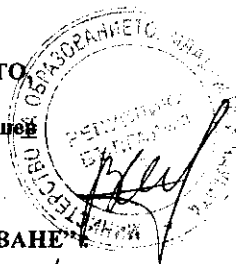
МИНИСТЕРСТВО НА ОБРАЗОВАНИЕТО
МЛАДЕЖТА И НАУКАТА :

/п/ чл.- кор. Николай Милошев

Вярно,

ДИРЕКТОР НА ДИРЕКЦИЯ
„КАНЦЕЛАРИЯ И
АДМИНИСТРАТИВНО ОБСЛУЖВАНЕ“

/ Красимира Коева /



МИНИСТЕРСТВО НА ОБРАЗОВАНИЕТО, МЛАДЕЖТА И НАУКАТА

НАЦИОНАЛНА ИЗПИТНА ПРОГРАМА

**ЗА ПРОВЕЖДАНЕ НА ДЪРЖАВНИ ИЗПИТИ ЗА ПРИДОБИВАНЕ НА
ТРЕТА СТЕПЕН НА ПРОФЕСИОНАЛНА КВАЛИФИКАЦИЯ**

	Код по СПОО	Наименование
Професионално направление	481	„КОМПЮТЪРНИ НАУКИ“
Професия	481020	„СИСТЕМЕН ПРОГРАМИСТ“
Специалност	4810201	„СИСТЕМНО ПРОГРАМИРАНЕ“

УТВЪРДЕНА СЪС ЗАПОВЕД № РД 09 - 553 / 07.05.2013 г.

София, 2013 година

I. ПРЕДНАЗНАЧЕНИЕ И ЦЕЛ НА ИЗПИТНАТА ПРОГРАМА

Националната изпитна програма е предназначена за организиране и провеждане на държавните изпити по теория и по практика на професията и специалността за придобиване на **трета** степен на професионална квалификация по професия код **481020 „Системен програмист“**, специалност код **4810201 „Системно програмиране“** от професионално направление код **481 „Компютърни науки“** от Списъка на професиите за професионално образование и обучение по чл. 6 от Закона за професионалното образование и обучение.

Целта на настоящата национална изпитна програма е да определи единни критерии за оценка на професионалните компетентности на обучаваните, изискващи се за придобиване на **трета** степен на професионална квалификация по изучаваната професия и специалност.

Националната изпитна програма е разработена във връзка с чл. 36 от Закона за професионалното образование и обучение (ЗПОО).

Държавните изпити по теория и по практика на професията и специалността се провеждат в съответствие с изискванията на ЗПОО и Наредба № 3 от 15.04.2003 г. за системата за оценяване.

II. СЪДЪРЖАНИЕ НА НАЦИОНАЛНАТА ИЗПИТНА ПРОГРАМА

Настоящата национална изпитна програма съдържа:

- 1. За държавния изпит по теория на професията и специалността:**
 - а. Изпитните теми с план-тезиса на учебното съдържание.
 - б. Критерии за оценяване.
- 2. За държавния изпит по практика на професията и специалността:**
 - а. Указания за съдържанието на индивидуалните практически задания.
 - б. Критерии за оценяване.
- 3. Система за оценяване.**
- 4. Препоръчителна литература.**
- 5. Приложения:**
 - а. Примерен изпитен билет за държавния изпит по теория на професията и специалността.
 - б. Примерно индивидуално практическо задание.

III. ДЪРЖАВЕН ИЗПИТ ПО ТЕОРИЯ НА ПРОФЕСИЯТА И СПЕЦИАЛНОСТТА

1. Изпитни теми с план-тезис на учебното съдържание

Посочените приложни задачи в изпитните теми са **примерни**. Комисията по оценяване на изпита по теория, назначена със заповед на директора на училището/ръководителя на обучаващата институция, може да модифицира примерите или да разработи нови варианти на приложни задачи.

Критериите за оценяване са в съответствие с посочените в Държавното образователно изискване (ДОИ) за придобиване на квалификация по професията „Програмист“, Наредба № 20 от 24 септември 2004 г.

Комисията по оценяване на изпита по теория определя за всеки критерий конкретни показатели, чрез които да се диференцира конкретният брой присъдени точки.

ИЗПИТНА ТЕМА 1. Скаларни типове данни в езика C++

План-тезис: Скаларни типове данни. Числов, символен и логически тип. Преобразуване на типове. Тип указател и тип псевдоним.

Необходима минимална конфигурация за инсталиране на езика за програмиране C++.

Трудово възнаграждение.

Правила за безопасност на работното място.

Примерни приложни задачи:

Задача 1.

Разчетете програмата, написана на C++, в която R1 и R2 са стойности на съпротивленията на резистори, а Rp – общото съпротивление на R1 и R2 при паралелно свързване на резисторите. Обяснете действието ѝ и модифицирайте програмния код така, че да се изчислява общото съпротивление RS на R1 и R2 при последователно свързване на резисторите.

```
#include <iostream>
using namespace std;
int main ()
{
    float R1, R2, Rp;
    cout << "\nPlease enter value of R1 ";
    cin >> R1;
    cout << "\nPlease enter value of R2 ";
    cin >> R2;
    if (R1 + R2 == 0)
    {
        cout << "\nError: Result cannot be calculated";
    }
    else
    {
        Rp = R1 * R2 / (R1 + R2);
        cout << "\nThe value is ";
        cout << Rp;
    }
    return 0; }
```

Задача 2.

Да се напише програма, която да пресмята по колко ученици от три класа, получили съответно общи първа, втора и трета награда за участие в състезание по програмиране, могат да отидат на екскурзия. Размерът на всяка обща награда и цената на екскурзията за 1 ученик (в лева и стотинки) са известни и се въвеждат от клавиатурата.

Дидактически материали: справочни материали за синтаксиса на типовете данни и операторите в C++.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява понятието тип данни. Прави класификация на вградените типове данни в C++. Дефинира и обяснява числов, символен и логически тип; множество от стойности, оператори и вградени функции; преобразуване на типове; тип указател и тип псевдоним в C++.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на езика за програмиране.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на C++. Разбира алгоритъма на предоставена програма и модифицира програмния код в съответствие със заданието.	12
	Задача 2. Разработва по задание програма с използване на скаларни типове данни.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – трудово възнаграждение.	3
5.	Познава правилата за осигуряване на безопасност на работното място.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
	Общ брой точки:	60

ИЗПИТНА ТЕМА 2. Съставни типове в C++ - масиви, символни низове и структури

План-тезис: Съставни типове данни. Масиви, многомерни масиви. Обработка на масиви с указатели. Символни низове – масиви от символи, клас string. Операции и вградени функции. Структури от данни, масиви от структури. Указатели към структури.

Необходима минимална конфигурация за инсталиране на езика за програмиране C++.

Прекратяване на трудов договор.

Основни ергономични изисквания към работните места за работа с видеодисплей.

Примерни приложни задачи:

Задача 1.

Разчетете програмата, написана на C++. Обяснете действието на програмата. Преобразувайте програмата, като създадете функция, която да извършва това действие и да бъде извиквана от главната функция (символният низ се въвежда в главната функция).

```
#include<iostream>
using namespace std;
```

```

void main()
{
char word[30], drow[30];
int w_length,i;
for (;;)
{
cout<<"\nEnter a word:( Enter a symbol for end)\n";
cin>>word;
w_length=strlen(word);
if (w_length!=1)
{
for (i=0;i<w_length;i++)
{
drow[i]=word[w_length-i-1];
}
for(i=0;i<w_length;i++)
{
cout<<drow[i];
}
}
else
{
cout<<endl;
break;
}
}
return;
}

```

Задача 2.

Всяка редица от равни числа в едномерен сортиран масив се нарича площадка. Да се напише програма, която намира началото и дължината на всички площадки в даден сортиран във възходящ ред едномерен масив `arr_sort[10]`, а ако няма площадки, да извежда съответно съобщение.

Дидактически материали: справочни материали за синтаксиса на типовете данни и операторите в C++.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява понятията тип данни и съставен тип данни. Дефинира и обяснява масиви и символни низове. Описва основните операции и вградени функции с масиви и символни низове. Дефинира и обяснява структури от данни, масиви от структури и указатели към структури.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на езика за програмиране.	3
3.	Решава приложни задачи: Задача 1. Разчита програма (или програмен фрагмент), написана на C++. Разбира алгоритъма на предоставена програма и модифицира програмния код в съответствие със заданието.	12

	Задача 2. Разработва по задание програма с използване на съставни типове данни.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – прекратяване на трудов договор.	3
5.	Познава основните ергономични изисквания към работните места за работа с видеодисплей.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
	Общ брой точки:	60

ИЗПИТНА ТЕМА 3. Основни оператори за управление на изчислителния процес в програмирането на C++

План-тезис: Основни оператори за управление на изчислителния процес в програмирането на C++ – оператор за присвояване, празен оператор, блок, условни оператори, оператори за цикъл. Примери.

Необходима минимална конфигурация за инсталиране на езика за програмиране C++.

Полагане на допълнителен труд по трудов договор.

Правила за безопасност на работното място.

Примерни приложни задачи:

Задача 1.

В зададения код на езика C++ открийте, документируйте и поправете грешките. Напишете кода на друг език за програмиране (*езикът за програмиране е от изучаваните Pascal, Visual Basic, JavaScript или Java*).

```

Include <iostream.h> ;
Int Mein ();
{
cout<<"n= ";
cin<<n
int n;
if (!cin)
{cout<<"Error. Bad input! \n";
Return 1;
}
If (n<=0){
Cout>>"Incorrect input! /n"; }
Return 1;
Int fact=1;
For (i=1, i<=n, i++)
Fact=fact*I;
Cout<<n<<"! = "<<fact<<"\n";
Return0;

```

Задача 2.

Да се напише програма, която по дадени x -реално и n -естествено число пресмята сумата. Въведените стойности и резултатите запишете във файл.

$$S = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \dots + \frac{x^n}{n!}$$

Дидактически материали: справочни материали за синтаксиса на типовете данни и операторите в C++.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява основните оператори за управление на изчислителния процес в програмирането на посочения език.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на езика за програмиране.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на C++. Разбира алгоритъма на предоставена програма и модифицира програмния код в съответствие със заданието.	12
	Задача 2. Разработва по задание програма с използване на оператори за изчисления и управление на изчислителния процес.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – полагане на допълнителен труд по трудов договор.	3
5.	Познава правилата за осигуряване на безопасност на работното място.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
Общ брой точки:		60

ИЗПИТНА ТЕМА 4. Функции. Параметри и област на действие на променливите. Връщане на стойност от функции

План-тезис:

Функции – същност и предимства. Деклариране, дефиниране и извикване на функция. Параметри на функция, подразбиращи се параметри и област на действие на променливите. Параметри-стойности и параметри-псевдоними. Масиви и указатели като параметри на функции. Връщане на стойност от функции. предефиниране на функции.

Необходима минимална конфигурация за инсталиране на езика за програмиране C++.

Видове срочни трудови договори.

Фактори, оказващи влияние върху работоспособността.

Примерни приложни задачи:

Задача 1.

Разчетете програмата, написана на C++. Обяснете действието на програмата, запишете как ще бъдат подредени числата в масива при $i=1$, $j=2$ и обяснете защо.

```
#include <iostream>
using namespace std;
const int N=5;
int a[N]={91,34,76,22,43};
```



```

void BubbleSort (int a[], int n)
{
    int i,j;
    int temp;
    for (i=0; i<n-1; i++)
    {
        for (j=1; j<n-i;j++)
        {
            if (a[j-1]>a[j])
            {
                temp=a[j-1];
                a[j-1]=a[j];
                a[j]=temp;}}
    }

int main()
{
BubbleSort (a,N);
for (int i=0; i<N; i++)
cout<<a[i]<<" ";
cout<<endl;
return 0;
}

```

Задача 2.

Да се напише програма на C++, която въвежда двумерен масив А [n, k] от цели числа от интервала [-1000, 1000]. Да се образува нова матрица В от членовете на матрицата А, като се създаде функция, която увеличава всеки нечетен елемент с 1, а всеки четен – с 2 (да се използват указатели към масив).

Дидактически материали: справочни материали за синтаксиса на типовете данни и операторите в C++.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява същност и предимства на функциите в C++, деклариране, дефиниране и извикване на функции, параметри на функции и област на действие на променливите, връщане на стойности от функции.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на езика за програмиране.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на C++. Разбира алгоритъма на предоставена програма и модифицира програмния код в съответствие със заданието.	12
	Задача 2. Разработва по задание програма с използване на функции.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – видове срочни трудови договори.	3
5.	Познава факторите, оказващи влияние върху работоспособността.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
	Общ брой точки:	60

ИЗПИТНА ТЕМА 5. Класове и обекти

План-тезис:

Същност на обектно-ориентираното програмиране. Класове и обекти. Дефиниране на класове и обекти в C++. Конструктори и деструктори. Динамична памет, създаване и разрушаване на обекти и масиви от обекти в динамичната памет.

Необходима минимална конфигурация за инсталиране на езика за програмиране C++.

Съдържание на индивидуалния трудов договор.

Долекарска помощ на пострадал от електрически удар.

Примерни приложни задачи:

Задача 1.

В зададения код на езика C++ открийте, документируйте и поправете грешките. Напишете кода на друг език за програмиране (*езикът за програмиране е от изучаваните Visual Basic, JavaScript или Java*).

```
Include <iostream >
Class CRectangle{
float: x, y
public:
CRectangle(float, float);
int Area (void)
{return (x*y);}
};
CRectangle:: CRectangle(int a, int b)
{
x=a; y=b;
}
Main
CRectangle Rect1(3,4);
cout << "The rectangle area is: " << rect1.area()<<endl;
}
```

Задача 2.

Да се дефинира клас „точка в равнината” с член-променливи – двете координати на точката и подходящи член-функции. Като се използва дефинираният клас, да се напише програма, която:

- въвежда n различни точки от равнината;
- намира разстоянието между произволни две точки;

$$d_{AB} = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}$$

- намира най-малкото разстояние.

Дидактически материали: справочни материали за синтаксиса на типовете данни и операторите в C++.

№	Критерии за оценяване	Максимален брой точки
1.	Обяснява същността на обектно-ориентираното програмиране и изброява принципите му. Дефинира и обяснява понятията клас и обект. Дефинира синтаксиса и семантиката на реализацията на класове и обекти в език за програмиране C++. Дефинира понятието динамична памет и обяснява създаването и разрушаването на обекти и масиви от обекти в динамичната памет.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на езика за програмиране C++.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на C++. Разбира алгоритъма на предоставена програма и модифицира програмния код в съответствие със заданието.	12
	Задача 2. Разработва по задание програма с използване на класове и обекти.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – съдържание на индивидуалния трудов договор.	3
5.	Познава изискванията за оказване на долекарска помощ на пострадал от електрически удар.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
	Общ брой точки:	60

ИЗПИТНА ТЕМА 6. Потоци и работа с файлове

План-тезис:

Организация на библиотеката `iostream` в C++. Потоци. Входно/изходни операции. Състояние на поток. Форматиране и манипулатори. Файлове, основни операции.

Необходима минимална конфигурация за инсталиране на езика за програмиране C++.

Съдържание на индивидуалния трудов договор.

Долекарска помощ на пострадал от изгаряне.

Примерни приложни задачи:

Задача 1.

Опишете действието на всеки от следващите програмни фрагменти. Сравнете ги и обяснете разликите.

```
#include <fstream>
using namespace std;
void main()
{
    char ch;
    ifstream fin("file2.txt", ios::in);
    ofstream fout("file1.txt", ios::out);
    while(fin >> ch)fout<< ch;
    cout<<endl;
}
```

```
#include <fstream>
using namespace std;
void main()
{
    char ch;
    ifstream fin("file2.txt", ios::in);
    ofstream fout("file1.txt", ios::out);
    while(fin.get(ch))fout << ch;
    cout<<endl;
}
```

Задача 2.

Да се напише програма на C++, която създава файл от низове с дължина не по-голяма

от 80 символа. За край на въвеждането да служи низът "end", който да не се включва във файла.

Дидактически материали: справочни материали за синтаксиса на типовете данни и операторите в C++.

№	Критерии за оценяване	Максимален брой точки
1.	Обяснява организация на библиотеката <code>iostream</code> в C++. Дефинира и обяснява потоци, входно/изходни операции, състояние на поток. Форматиране и манипулатори. Файлове, основни операции.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на езика за програмиране.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на C++. Разбира алгоритъма на предоставена програма и сравнява различни варианти на програмния код.	12
	Задача 2. Разработва по задание програма с използване на потоци и файлови операции.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – съдържание на индивидуалния трудов договор.	3
5.	Познава изискванията за оказване на долекарска помощ на пострадал от електрически удар.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
Общ брой точки:		60

ИЗПИТНА ТЕМА 7. Линейни динамични структури от данни в езика за програмиране C++ – стек и опашка

План-тезис:

Линейни динамични структури от данни в езика за програмиране C++ – стек, опашка. Общо описание, основни операции, реализация. Класове `stack` и `queue` от STL библиотеката. Необходима минимална конфигурация за инсталиране на езика за програмиране C++. Разлика между индивидуален и колективен трудов договор. Фактори, оказващи влияние върху работоспособността.

Примерни приложни задачи:

Задача 1.

Опишете действието на зададената програма. Преобразувайте я, като вместо опашка използвате стек, и обяснете очаквания резултат.

```
#include <iostream>
#include <queue>
using namespace std;
int main()
{
    int i;
    string ball;
    queue <string> balls;
    for(i = 0; i < 5; i++)
```

```

{
    cout << "Push ball: ";
    cin >> ball;
    balls.push(ball);
}

while(!balls.empty())
{
    ball = balls.front();
    if(ball != "white")
    {
        cout << ball << endl;
        balls.pop();
        if(balls.empty())
        {
            cout << "There is no white ball!\n";
            break;
        }
    } else
    {
        cout << "White ball! => Stop Popping.\n";
        break;
    }
}
return 0; }

```

Задача 2.

Да се напише програма, която извежда двоичното представяне на естествено число. За обръщане на двоичното представяне в прав ред да се използва стек.

Дидактически материали: справочни материали за синтаксиса на типовете данни и операторите в C++.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява линейни динамични структури от данни в езика за програмиране C++ – стек, опашка. Общо описание, основни операции, реализация. Класове stack и queue от STL библиотеката.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на езика за програмиране.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на C++. Разбира алгоритъма на предоставена програма и модифицира програмния код в съответствие със заданието.	12
	Задача 2. Разработва по задание програма с използване на линейни динамични структури от данни.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – разлика между индивидуален и колективен трудов договор .	3
5.	Познава факторите, оказващи влияние върху работоспособността.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
	Общ брой точки:	60

ИЗПИТНА ТЕМА 8. Проектиране на релационни бази от данни

План-тезис: Същност на релационните бази от данни. Проектиране на релационни бази от данни. Анализ на обектите и дейностите. Създаване на концептуален модел. Създаване на логически модел - определяне на таблиците, полетата и ключовите полета, нормализиране на релациите, определяне на връзките между таблиците.

Необходима минимална конфигурация за инсталиране на MS Access.

Полагане на допълнителен труд по трудов договор.

Методи и средства за гасене на пожари.

Примерни приложни задачи:

Задача 1.

В зададения програмен фрагмент на C++ открийте, документируйте и поправете грешките. Напишете кода на друг език за програмиране (*езикът за програмиране е от изучаваните Pascal, Visual Basic, или Java*).

```
Const int NUM=5;
Struct student {
    Int nomer;
    Char name[26];
    Double marks[NUM];
};
Void read_student(student& s)
{
    Cout<<"nomer: ";
    Cin>>s.nomer;
    Char p[100];
    Cin.getline(p,100);
    Cout<<"name ";
    Cin.getline(s.ime, 26);
    For (i=0, i<=NUM-1, i++)
    {
        Cout<<i<<"-th mark: ";
        Cin<<s.marks[i];
    }
}
```

Задача 2.

Проектирайте релационна база от данни „Училище”, която да съдържа информация за учениците (ЕГН, име, фамилия, пол, град, адрес, телефон, родител) и годишните оценки по предметите, които изучават. Начертайте концептуалния и логическия модел. Определете таблиците, полетата, ключовите полета и връзките между таблиците. Нормализирайте релационната схема в трета нормална форма.

Дидактически материали: справочни материали за синтаксиса на типовете данни и операторите на изучаваните програмни езици.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява същност на релационните бази от данни. Обяснява процеса на проектиране на релационни бази от данни. Анализ на обектите и дейностите. Създаване на концептуален модел. Създаване на логически модел – определяне на таблиците, полетата и ключовите полета, нормализиране на релациите, определяне на връзките между таблиците.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на системата за управление на бази от данни MS Access.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на изучаван език за програмиране. Открива, документира и коригира синтактични и логически грешки в предоставения програмен код. Разбира алгоритъма на предоставена програма и пренаписва програмата на друг изучаван програмен език.	12
	Задача 2. Проектира релационна база от данни по задание.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – полагане на допълнителен труд по трудов договор.	3
5.	Познава методите и средствата за гасене на пожари.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
	Общ брой точки:	60

ИЗПИТНА ТЕМА 9. Създаване на база от данни и таблици в MS Access

План-тезис:

Система за управление на база данни (СУБД). Характеристика на СУБД MS Access. Създаване на база данни и таблици в Access. Осигуряване коректност на данните (data integrity) чрез типа на данните, входни маски, подразбиращи се стойности, валидиращи правила, справочни списъци, задължително попълване, индексирание. Описание на типовете данни. Създаване на връзки между таблиците. Типове връзки, свойства на връзките.

Необходима минимална конфигурация за инсталиране на системата за управление на бази от данни MS Access.

Напишете поне 3 задължения на работодателите за осигуряване на здравословни и безопасни условия на труд.

Основни ергономични изисквания към работните места за работа с видеодисплей.

Примерни приложни задачи:

Задача 1.

В зададения код на езика C++ открийте, документирайте и поправете грешките. Напишете кода на друг език за програмиране (езикът за програмиране е от изучаваните Pascal, Visual Basic, JavaScript или Java).

```

Include iostream.h;
Include iomanip.h;
{
int a[100]
cout<<"n= ";
cin<<n;
int n;
if (!cin)
{cout<<"Error. Bad input! \n";
Return 1; }
If (n<=0) {
Cout<<"Incorrect input! \n"; }
Return 1;
Int i;
For (i=0; i<=n-2; i++);
{int min= a[i]; Int k=i;
For (int j = i+1; j<=n-1; j--)
If (a[j] > min)
{
min = a[j];
k = j;
}
Int x = a[i]; a[i] = a[k]; a[k] = x; }
For (i = 0; i <= n-1; i++)
cout<<setw (10)<<a[i] <<"\n";
Return 0; }

```

Задача 2.

В база данни „Училище” има три таблици: Ученици, Предмети и Оценки. Връзката между Ученици и Предмети е много към много и се осъществява чрез таблицата Оценки, която съдържа три полета: идентификатор на ученика – външен ключ, идентификатор на предмета – външен ключ и оценката на ученика по съответния предмет. Таблицата Ученици съдържа полетата: Код, Име, Презиме, Фамилия, Дата на раждане, Пол, Адрес, Телефон, Клас, Паралелка, Бележки. Таблицата Предмети съдържа полетата: Код, Наименование.

Да се зададат подходящи свойства на полетата на таблиците от БД „Училище”, така че да се осигури коректност на въвежданите данни. Да се начертае реляционната схема на базата данни.

Дидактически материали: справочни материали за синтаксиса на типовете данни и операторите на изучаваните програмни езици, заснет прозорец на таблица в режим на проектиране заедно с панела за свойствата на полетата.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява система за управление на база данни (СУБД). Изброява характеристики на СУБД MS Access. Описва създаване на база данни и таблици в Access. Обяснява осигуряване коректност (цялостност) на данните (data integrity) чрез типа на данните, входни маски, подразбиращи се стойности, валидиращи правила, справочни списъци, задължително попълване, индексирание. Изброява и описва типовете данни в MS Access. Описва създаване на връзки между таблиците. Обяснява типове връзки, свойства на връзките.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на системата за управление на бази от данни MS Access.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на изучаван език за програмиране. Открива, документира и коригира синтактични и логически грешки в предоставения програмен код. Разбира алгоритъма на предоставена програма и пренаписва програмата на друг изучаван програмен език.	12
	Задача 2. Задава подходящи свойства на полетата на таблиците от БД, така че да се осигури коректност на въвежданите данни. Чертае релационната схема на базата данни – таблиците с полетата и ключовите полета и връзките между тях.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – задължения на работодателите за осигуряване на здравословни и безопасни условия на труд.	3
5.	Познава основните ергономични изисквания към работните места за работа с видеодисплей.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
	Общ брой точки:	60

ИЗПИТНА ТЕМА 10. Формуляри в MS Access

План-тезис:

Формуляри – същност и предназначение. Елементи на формуляра. Създаване на формуляри чрез асистент (wizard) и ръчно. Главен и подчинен формуляр. Въвеждане на данните чрез формуляри. Създаване на управляващ панел (switchboard) чрез Switchboard manager.

Необходима минимална конфигурация за инсталиране на MS Access.

Полагане на допълнителен труд по трудов договор.

Вредни фактори за здравето и безопасността за професията „Системен програмист“.

Примерни приложни задачи:

Задача 1.

В зададения код на езика C++ открийте, документирайте и поправете грешките. Напишете кода на друг език за програмиране (езикът за програмиране е от изучаваните Pascal, Visual Basic, JavaScript или Java).

```

Includ <iostriam.h> ;
Includ <iomanip.h> ;
Int Main ()
{
enum weekday{SUNDAY, MONDAY, TUESDAY, WEDNESDAY, THURSDAY,
FRIDAY, SATURDAY};
Weekday d;
Double s=0;
For[d=SUNDAY, d<=SATURDAY, d=(Weekday)(d+1)]
{
Double t;
Cout<<>>"temperatura=" ";
Cin<<temperatura;
}
S+= temperatura;
caut<<setprecision(2)<<setiosflags(ios::fixed);
caut<<setw(10)<<s/7<<"\n";
Return 0;

```

Задача 2.

В база данни „Училище” има три таблици: Ученици, Предмети и Оценки. Връзката между Ученици и Предмети е много към много и се осъществява чрез таблицата Оценки, която съдържа три полета: идентификатор на ученика – външен ключ, идентификатор на предмета – външен ключ и оценката на ученика по съответния предмет. Таблицата Ученици съдържа полетата: Код, Име, Презиме, Фамилия, Дата на раждане, Пол, Адрес, Телефон, Клас, Паралелка, Бележки. Таблицата Предмети съдържа полетата: Код, Наименование.

В базата данни „Училище” да се опише създаването на формуляр за въвеждане на данните в таблиците, като се използват подходящи контроли за съответните полета – текстови полета, падащи списъци, полета за отметки и др.

Дидактически материали: справочни материали за синтаксиса на типовете данни и операторите на изучаваните програмни езици, реляционна схема на БД „Училище”, прозорец на формуляр в изглед за проектиране заедно с кутията с инструменти.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява същност и предназначение на формулярите (формите). Описва елементите на формуляра. Описва създаване на формуляри чрез асистент (wizard) и ръчно. Описва създаване на главен и подчинен формуляр и взаимодействието между тях. Обяснява въвеждане на данните чрез формуляри. Описва създаване на управляващ панел (switchboard) чрез Switchboard manager.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на системата за управление на бази от данни MS Access.	3
3.	Решава приложни задачи:	

	Задача 1. Разчита програма (или програмен фрагмент), написана на изучаван език за програмиране. Открива, документира и коригира синтактични и логически грешки в предоставения програмен код. Разбира алгоритъма на предоставена програма и пренаписва програмата на друг изучаван програмен език.	12
	Задача 2. Описва създаването на формуляр за въвеждане на данните в таблиците на зададена база от данни.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – полагане на допълнителен труд по трудов договор.	3
5.	Познава вредните фактори за здравето и безопасността за професията „Системен програмист“.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
	Общ брой точки:	60

ИЗПИТНА ТЕМА 11. Откриване на конкретна информация в базата данни. Сортиране, филтриране и заявки (queries)

План-тезис: Сортиране и филтриране на информация, знакове за заместване. Същност и предназначение на заявките (запитванията). Видове заявки. Видове заявки за селектиране. Създаване на обикновени заявки, заявки с параметър и обобщени заявки в MS Access. Изчислителни полета, използване на Expression Builder.

Необходима минимална конфигурация за инсталиране на системата за управление на бази от данни MS Access.

Прекратяване на трудов договор.

Фактори, оказващи влияние върху работоспособността.

Примерни приложни задачи:

Задача 1.

В зададения код на езика C++ открийте, документирайте и поправете грешките. Напишете кода на друг език за програмиране (*езикът за програмиране е от изучаваните Pascal, Visual Basic, JavaScript или Java*).

```
#Include (iostream.h);
Double CircleArea[double r]
{
    Const double PI;
    PI = 3.14159265;
    Return Pi*R*R;
}

void Main;
{
    cout >> 'Въведете радиус на кръга?';
    cin >> Radius;
    cout << "Лицето на кръга с радиус " << Radius << " е " << CircleArea(Radius) << endl;
    return 0;
}
```

}

Задача 2.

В база данни „Училище” има три таблици: Ученици, Предмети и Оценки. Връзката между Ученици и Предмети е много към много и се осъществява чрез таблицата Оценки, която съдържа три полета: идентификатор на ученика – външен ключ, идентификатор на предмета – външен ключ и оценката на ученика по съответния предмет. Таблицата Ученици съдържа полетата: Код, Име, Презиме, Фамилия, Дата на раждане, Пол, Адрес, Телефон, Клас, Паралелка, Бележки. Таблицата Предмети съдържа полетата: Код, Наименование.

а) Да се проектира в QBE таблицата параметрична заявка (query), която да показва името, фамилията и оценките по математика на учениците, които са мъже и са родени преди дата, зададена като параметър.

б) Да се създаде заявка, базирана на таблицата Ученици, която да съдържа следните изчислителни полета:

Име_Ученик – обединение на Име, Презиме, Фамилия, с интервали помежду им;

Възраст – разлика между текущата дата Date() и датата на раждане

Пол_Текст - ако Пол от таблицата е Yes – връща „мъж”, ако е No - връща „жена”

Описание на дидактическите материали: Справочни материали за синтаксиса на типовете данни и операторите на изучаваните програмни езици. Схема на празна QBE таблица на заявка в проектен изглед заедно със схема на базовите таблици от БД „Училище”.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява сортиране и филтриране на информация, знакове за заместване. Същност и предназначение на заявките (запитванията). Видове заявки. Видове заявки за селектиране. Създаване на обикновени заявки, заявки с параметър и обобщени заявки в MS Access. Изчислителни полета, използване на Expression Builder.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на системата за управление на бази от данни MS Access.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на изучаван език за програмиране. Открива, документира и коригира синтактични и логически грешки в предоставения програмен код. Разбира алгоритъма на предоставена програма и пренаписва програмата на друг изучаван програмен език.	12
	Задача 2. Описва създаването на заявки за извличане на информация от конкретна база от данни по зададени критерии.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – прекратяване на трудов договор.	3
5.	Познава факторите, оказващи влияние върху работоспособността.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
	Общ брой точки:	60

ИЗПИТНА ТЕМА 12. Проектиране на Web страница

План-тезис:

Етапи на проектиране на Web страница. Дейности във всеки един етап.
Необходима минимална конфигурация за инсталиране на изучавания web редактор.
Видове срочни трудови договори.
Правила за безопасност на работното място.

Примерни приложни задачи:

Задача 1.

В зададения код на езика C++ открийте, документируйте и поправете грешките. Напишете кода на друг език за програмиране (*езикът за програмиране е от изучаваните Pascal, Visual Basic, JavaScript или Java*).

```
Include iostream.h;
Include iomanip.h;
{
int a[100]
cout<<"n= ";
cin<<n;
int n;
if (!cin)
{cout<<"Error. Bad input! \n";
Return 1; }
If (n<=0) {
Cout<<"Incorrect input! \n"; }
Return 1;
Int i;
For (i=0; i<=n-2; i++);
{int min= a[i]; Int k=i;
For (int j = i+1; j<=n-1; j--)
If (a[j] > min)
{
min = a[j];
k = j;
}
Int x = a[i]; a[i] = a[k]; a[k] = x; }
For (i = 0; I <= n-1; i++)
cout<<setw (10)<<a[i] <<"\n";
Return 0; }
```

Задача 2.

Да се опишат етапите при проектиране на сайт на модна агенция. Да се изброят целите и конкретните дейности за всеки етап. Да се посочат продуктите, които трябва да се изготвят в края на всеки етап.

Описание на дидактическите материали: справочни материали за синтаксиса на типовете данни и операторите на изучаваните програмни езици.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява етапи на проектиране на Web страница. Дейности във всеки един етап. Документи в края на всеки етап.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на изучавания web редактор.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на изучаван език за програмиране. Открива, документира и коригира синтактични и логически грешки в предоставения програмен код. Разбира алгоритъма на предоставена програма и пренаписва програмата на друг изучаван програмен език.	12
	Задача 2. Описва проектирането на Web страница.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – видове срочни трудови договори.	3
5.	Познава правилата за осигуряване на безопасност на работното място.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
	Общ брой точки:	60

ИЗПИТНА ТЕМА 13. Основи на HTML

План-тезис:

Същност и предназначение на езика HTML. Структура и съдържание на HTML документ; метаданни. Тагове (елементи) и атрибути. Основни HTML тагове – за форматиране на текст, списъци, хоризонтални линии, хипервръзки, изображения, таблици, рамки (фреймове), форми. Начини за дефиниране на цвят. Нови елементи в HTML 5.

Необходима минимална конфигурация за инсталиране на изучавания web редактор.

Съдържание на индивидуалния трудов договор.

Долекарска помощ на пострадал от електрически удар.

Примерни приложни задачи:

Задача 1.

В зададения програмен фрагмент на C++ открийте, документирайте и поправете грешките. Напишете кода на друг език за програмиране (*езикът за програмиране е от изучаваните Pascal, Visual Basic, или Java*).

```
Const int NUM=5;
Struct student {
    Int nomer;
    Char name[26];
    Double marks[NUM];
};
Void read_student(student& s)
{
    Cout<<"nomer: ";
    Cin>>s.nomer;
    Char p[100];
    Cin.getline(p,100);
    Cout<<"name ";
    Cin.getline(s.ime, 26);
    For (i=0, i<=NUM-1, i++)
    {
        Cout<<i<<"-th mark: ";
        Cin<<s.marks[i];
    }
}
```

Задача 2.

Създайте лична web страница, като напишете HTML код. В страницата включете заглавна част със заглавие, кодова таблица (charset) – кирилица, ключови думи. В тялото на страницата използвайте цвят или картинка за фон, различни видове форматиране на текст, подредени и неподредени списъци, хоризонтални линии, хипервръзки, изображения, таблици.

Описание на дидактическите материали: справочни материали за синтаксиса на типовете данни и операторите на изучаваните програмни езици.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява същност и предназначение на езика HTML, структура и съдържание на HTML документ; метаданни. Обяснява същност и синтаксис на тагове (елементи) и атрибути. Дефинира основни HTML тагове – за форматиране на текст, списъци, хоризонтални линии, хипервръзки, изображения, таблици, рамки (фреймове), форми. Изброява начини за дефиниране на цвят. Нови елементи в HTML 5.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на изучавания web редактор.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на изучаван език за програмиране. Открива, документира и коригира синтактични и логически грешки в предоставения програмен код. Разбира алгоритъма на предоставена програма и пренаписва програмата на друг изучаван програмен език.	12
	Задача 2. Създава web страница чрез HTML код.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – съдържание на индивидуалния трудов договор.	3
5.	Познава правилата за оказване на долекарска помощ на пострадал от електрически удар.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
	Общ брой точки:	60

ИЗПИТНА ТЕМА 14. Каскадни набори от стилове CSS

План-тезис:

Същност и предимства на CSS. Синтаксис. Създаване и употреба на CSS стилове. Начини за вмъкване на CSS в HTML документ: външен и вътрешен набор от стилове; вграден стил. Основни CSS свойства. Псевдокласове.

Необходима минимална конфигурация за инсталиране на изучавания web редактор.

Съдържание на индивидуалния трудов договор.

Правила за безопасност на работното място.

Примерни приложни задачи:

Задача 1.

В зададения код на C++ открийте, документирайте и поправете грешките. Напишете кода на друг език за програмиране (*езикът за програмиране е от изучаваните Pascal, Visual Basic, JavaScript или Java*).

```
#Include (iostream.h);
```

```
Double CircleArea[double r]
```

```
{
```

```
Const doble PI;
```

```
PI = 3.14159265;
```

```
Return Pi*R*R;
```



```

}

void Main;
    cout >> "Въведете радиус на кръга?";
    cin >> Radius;
    cout << "Лицето на кръга с радиус " << Radius << " е " << CircleArea(Radius) << endl;
    return 0;
}

```

Задача 2.

Обяснете показания CSS код. Добавете стил за тага h1 със следните свойства: семейство на шрифта: "Times New Roman", Times, serif ; размер на шрифта: 22px; тежест на шрифта: bold; стил на шрифта: italic; декорация: подчертан; цвят: кафяв; подравняване: ляво.

```

body
{
    background-color: gray;
}
p, br, div, span
{
    font-family : Verdana, Arial, Helvetica;
    font-size : 12px;
    text-decoration: none;
    color: #000000;
    line-height:20%;
    width:90%;
}

ul.a {
    list-style-type:circle;
    margin-top:3px;
    margin-left:20px;
}
table, th, td
{
    border: 1px solid black;
}
.TableText
{
    font-family : Verdana, Arial, Helvetica;
    font-size : 12px;
}
a:link {color:#FF0000;} /* unvisited link */
a:visited {color:#00FF00;} /* visited link */
a:hover {color:#FF00FF;} /* mouse over link */
a:active {color:#0000FF;}

```

Описание на дидактическите материали: Справочни материали за синтаксиса на типовете данни и операторите на изучаваните програмни езици.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява същност и предимства на CSS. Синтаксис на CSS. Създаване и употреба на CSS стилове. Начини за вмъкване на CSS в HTML документ: външен и вътрешен набор от стилове; вграден стил. Основни CSS свойства. Псевдокласове.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на изучавания web редактор.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на изучаван език за програмиране. Открива, документира и коригира синтактични и логически грешки в предоставения програмен код. Разбира алгоритъма на предоставена програма и пренаписва програмата на друг изучаван програмен език.	12
	Задача 2. Разчита и обяснява показан CSS код. Добавя стил по задание.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – съдържание на индивидуалния трудов договор.	3
5.	Познава правилата за безопасност на работното място.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
Общ брой точки:		60

ИЗПИТНА ТЕМА 15. Асемблер и Макроасемблер - I

План-тезис:

Асемблер и Макроасемблер – двоична аритметика, представяне на данните и инструкциите в различен формат, кодовите таблици, програми, стек, прекъсвания, видовете регистри, организация на паметта и методи за адресиране.

Необходима минимална конфигурация за инсталиране на изучавания компилатор за асемблер.

Разлика между индивидуален и колективен трудов договор.

Вредни фактори за здравето и безопасността за професията „Системен програмист“.

Примерни приложни задачи:

Задача 1.

В зададения код открийте, документирайте и поправете грешките. Напишете кода на друг език за програмиране (*езикът за програмиране е от изучаваните Pascal, Visual Basic, JavaScript или Java*).

```

Include <iostream.h>
Int Mein;
{
    cout<<"n= ";
    cin<<n;
    int n;
    if (!cin)
    {cout<<" Error. Bad input! /n ";

```

```

Return 1;
}
If (n<=0)
{
Cout<< ' Incorrect input! \n ';
}
Return 1;
Int fact=1;
For (int i=1; i<=n; i++)
Fact=fact*I;
Cout<<n<<"! = "<<fact<<"\n";
Return 0.

```

Задача 2.

Обяснете показаната програма, написана на Асемблер.

```

code_segment segment
assume cs:code_segment
start: mov cx, 256d ; character to output
      mov dl, 00h
      repeat:
            mov ah, 02h ; 21,02h display output
            int 21h ; character in DL
            inc dl
            loop repeat
end_program:
      mov ah,4ch ;terminate in DOS
      int 21h
code_segment ends
end start

```

Описание на дидактическите материали: справочни материали за синтаксиса на типовете данни и операторите на изучаваните програмни езици.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява Асемблер и Макроасемблер - двоична аритметика, представяне на данните и инструкциите в различен формат, кодовите таблици, програми, стек, прекъсвания, видовете регистри, организация на паметта и методи за адресиране.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на изучавания компилатор за асемблер.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на изучаван език за програмиране. Открива, документира и коригира синтактични и логически грешки в предоставения програмен код. Разбира алгоритъма на предоставена програма и пренаписва програмата на друг изучаван програмен език.	12
	Задача 2. Разчита и обяснява показан код на Асемблер.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – разлика между индивидуален и колективен трудов договор.	3

5.	Познава вредните фактори за здравето и безопасността за професията „Системен програмист“.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
Общ брой точки:		60

ИЗПИТНА ТЕМА 16. Асемблер и Макроасемблер - II

План-тезис:

Асемблер и Макроасемблер – видове регистри, флагове, вектори на прекъсване, сегменти и директиви, видове инструкции, управление и предаване на управлението на микропроцесора, аргументи на макросите, макродирективи.

Необходима минимална конфигурация за инсталиране на изучавания компилатор за асемблер.

Съдържание на индивидуалния трудов договор.

Долекарска помощ на пострадал от електрически удар.

Примерни приложни задачи:

Задача 1.

В зададения програмен фрагмент на C++ открийте, документируйте и поправете грешките. Напишете кода на друг език за програмиране (*езикът за програмиране е от изучаваните Pascal, Visual Basic, или Java*).

```

Const int NUM=5;
Struct student {
    Int nomer;
    Char name[26];
    Double marks[NUM];
};
Void read_student(student& s)
{
    Cout<<"nomer: ";
    Cin>>s.nomer;
    Char p[100];
    Cin.getline(p,100);
    Cout<<"name ";
    Cin.getline(s.ime, 26);
    For (i=0, i<=NUM-1, i++)
    {
        Cout<<i<<"-th mark: ";
        Cin<<s.marks[i];
    }
}

```

Задача 2.

Обяснете показаната програма, написана на Асемблер.

```
.model small
.stack
.code
start:
print_ascii proc
    mov dl,00h
    mov cx,256
print_loop:
    call write_char
    inc dl
    loop print_loop

    mov ah,4Ch
    int 21h
print_ascii endp

write_char proc
    mov ah,02h
    int 21h
    ret
write_char endp
end start
```

Описание на дидактическите материали: справочни материали за синтаксиса на типовете данни и операторите на изучаваните програмни езици.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява Асемблер и Макроасемблер – видове регистри, флагове, вектори на прекъсване, сегменти и директиви, видове инструкции, управление и предаване на управлението на микропроцесора, аргументи на макросите, макродирективи.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на изучавания компилатор за асемблер.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на изучаван език за програмиране. Открива, документира и коригира синтактични и логически грешки в предоставения програмен код. Разбира алгоритъма на предоставена програма и пренаписва програмата на друг изучаван програмен език.	12
	Задача 2. Разчита и обяснява показан код на Асемблер.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – съдържание на индивидуалния трудов договор.	3

5.	Познава правилата за оказване на долекарска помощ на пострадал от електрически удар.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
Общ брой точки:		60

Комисията по провеждане и оценяване на изпита по теория, назначена със заповед на директора на училището/ръководителя на обучаващата институция, подготвя билет за варианта на Изпитна тема 17, който съответства на учебната програма по Вградени управляващи системи, по която е извършено обучението.

ИЗПИТНА ТЕМА 17а. Обекти и класове в езика JAVA

План-тезис:

Дефиниране на класове в Java. Елементи на класа. Модификатори за достъп. Конструктори. Създаване на обекти от даден клас. Достъп до полета на обект. Извикване на методи на обект. Ключовата дума this.

Необходима минимална конфигурация за инсталиране на езика за програмиране Java.

Разлика между индивидуален и колективен трудов договор.

Фактори, оказващи влияние върху работоспособността.

Примерни приложни задачи:

Задача 1.

В зададения код на Java открийте, документирайте и поправете грешките. В коригираната програма напишете set и get методите за модифициране и извличане на страните на триъгълника a,b,c.

```
#import Java.util.scanner;
public Class Triangle {
    private Double a,b,c;
    private string color;
    private Triangle{
        This.a=10D;
        This.b=10d;
        This.c=10d;
        This.color="blue";
    }
    public triangle(A; B; C; String Color){
        this.a=A
        this.b=B
        this.c=C
        this.color=Color;
    }
    private String getColor(){
        return this.color;
    }
    public void setColor(String Color){
```

```

        this.color=Color;
    }
    private void print(){
        double p=this.a+this.b+this.c;
        System.out.printf("Триъгълникът със страни %f,%f,%f има обиколка:
%d",this.a,this.b,this.c,p);
    }

```

```

Public static void main(String[] args) {
    scanner INPUT=New Scanner(system.in);
    Triangle TR1=new Triangle();
    System.in.print('Въведете трите страни на триъгълника: ');
    TR1.setA(input.nextDouble());
    TR1.setB(input.nextDouble());
    TR1.setC(input.nextDouble());
    TR1.print;
}

```

Задача 2. Напишете програмния код на задача 1 на езика за програмиране C++.

Описание на дидактическите материали: Справочни материали за синтаксиса на типовете данни и операторите на изучаваните програмни езици.

№	Критерии за оценяване	Максимален брой точки
1.	Обяснява дефиниране на класове в Java. Елементи на класа. Модификатори за достъп. Конструктори. Създаване на обекти от даден клас. Достъп до полета на обект. Извикване на методи на обект. Ключовата дума this.	24
2.	Дефинира необходима минимална конфигурация за инсталиране на езика за програмиране Java.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на езика за програмиране Java. Открива, документира и коригира синтактични и логически грешки в предоставения програмен код. Дописва програмен код.	12
	Задача 2. Разбира алгоритъма на предоставената програма и пренаписва програмата от Java на C++.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – разлика между индивидуален и колективен трудов договор.	3
5.	Познава факторите, оказващи влияние върху работоспособността.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
Общ брой точки:		60

ИЗПИТНА ТЕМА 176. Основи на програмирането на Ардуино

План-тезис:

Същност на платформата Ардуино. Основи на програмирането на Ардуино: структура

на програма за платформата Ардуино на езика Processing, синтаксис на програмния код, символи, идентификатори, коментари. Типове данни и променливи – локални и глобални. Константи. Оператори – за присвояване, аритметични, за сравнение, логически. Функции. Вградени функции.

Необходима минимална конфигурация за работа с платформата Ардуино.

Разлика между индивидуален и колективен трудов договор.

Фактори, оказващи влияние върху работоспособността.

Примерни приложни задачи:

Задача 1.

В зададения код на езика C++ открийте, документируйте и поправете грешките. Напишете кода на друг език за програмиране (*езикът за програмиране е от изучаваните Pascal, Visual Basic, JavaScript или Java*).

```
Include iostream.h;
Include iomanip.h;
{
int a[100]
cout<<"n= ";
cin<<n;
int n;
if (!cin)
{cout<<"Error. Bad input! \n";
Return 1; }
If (n<=0) {
Cout<<"Incorrect input! \n"; }
Return 1;
Int i;
For (i=0; i<=n-2; i++);
{int min= a[i]; Int k=i;
For (int j = i+1; j<=n-1; j--)
If (a[j] > min)
{
min = a[j];
k = j;
}
Int x = a[i]; a[i] = a[k]; a[k] = x; }
For (i = 0; i <= n-1; i++)
cout<<setw (10)<<a[i] <<"\n";
Return 0; }
```

Задача 2.

Да се състави програма за Ардуино, която управлява електронна схема, съдържаща бутон, свързан към Pin2, и светодиод - към Pin13. При натиснат бутон изводът (pin) 2 да отчете ниво HIGH и да включи светодиода така, че той да мигне 10 пъти, като свети по 1 секунда и е угасен по 2 секунди.

Описание на дидактическите материали: справочни материали за синтаксиса на типовете данни и операторите на изучаваните програмни езици.

№	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява същност на платформата Ардуино, структура на програма за Ардуино, синтаксис на програмния код, символи, идентификатори, коментари. Типове данни и променливи – локални и глобални. Константи. Оператори – за присвояване, аритметични, за сравнение, логически. Функции. Вградени функции.	24
2.	Дефинира необходимата минимална конфигурация за работа с платформата Ардуино.	3
3.	Решава приложни задачи:	
	Задача 1. Разчита програма (или програмен фрагмент), написана на изучаван език за програмиране. Открива, документира и коригира синтактични и логически грешки в предоставения програмен код. Разбира алгоритъма на предоставена програма и пренаписва програмата на друг изучаван програмен език.	12
	Задача 2. Разработва програма за Ардуино за управление на хардуер със зададено поведение.	12
4.	Владее основните теоретични знания в областта на трудовоправните закони – разлика между индивидуален и колективен трудов договор.	3
5.	Познава факторите, оказващи влияние върху работоспособността.	3
6.	Знае основната професионална терминология в комуникацията на чужд език.	3
	Общ брой точки:	60

IV. ДЪРЖАВЕН ИЗПИТ ПО ПРАКТИКА НА ПРОФЕСИЯТА И СПЕЦИАЛНОСТТА

1. Указания за съдържанието на индивидуалните практически задания

Чрез държавния изпит по практика на професията и специалността се проверяват и оценяват професионалните умения и компетентности на обучаваните, отговарящи на трета степен на професионална квалификация.

Изпитът по практика на професията и специалността се състои в:

- избор на компоненти, модули, хардуерни и софтуерни средства съобразно практическото задание;
- инсталиране, поддържане и администриране на операционни системи и приложен софтуер по конкретно задание;
- писане на програмен код на избран език за програмиране от обучавания, създаване на конкретна база от данни, web сайт или създаване на вграден управляващ софтуер. Тестване на програмите.

Индивидуалните практически задания се разработват от комисия, назначена със заповед на директора/ръководителя на обучаващата институция.

Индивидуалното изпитно задание съдържа пълното наименование на училището/обучаващата институция, празни редове за попълване имената на обучавания, квалификационната форма, началната дата и началния час на изпита, краен срок на изпита – дата и час, темата на индивидуалното практическо задание и изискванията към крайния резултат от изпълнението на заданието. По решение на комисията могат да се дадат допълнителни указания, които да подпомогнат обучавания при изпълнение на индивидуалното практическо задание.

Индивидуалните практически задания се изготвят от комисията за провеждане и оценяване на изпита по практика на професията и специалността в училището/обучаващата институция. Броят на изготвените задания трябва да бъде поне с един повече от броя на явяващите се в деня на изпита. Всеки обучаван изтегля индивидуалното си практическо задание, в което веднага саморъчно написва трите си имена.

2. Критерии за оценяване

За всяко индивидуално практическо задание комисията по провеждане и оценяване на изпита по практика разработва критерии за оценяване и съответните показатели. Посочва се максималният брой точки, които се поставят при пълно вярно и точно изпълнение на показателите.

Критериите за оценяване са в съответствие с посочените в Държавното образователно изискване (ДОИ) за придобиване на квалификация по професията „Системен програмист“, Наредба № 20 от 24 септември 2004 г.

№	КРИТЕРИИ	ПОКАЗАТЕЛИ	Макс. брой точки
1.	Проверка и оценяване на професионалните компетенции в следните тематични области:		48
	Поддържане и тестване на КС, ПУ, мрежови хардуер и съответен софтуер.	- избор на компоненти, модули, хардуерни и софтуерни средства съобразно практическото задание	
	Инсталиране, конфигуриране и поддържане на операционни системи и приложен софтуер.	- инсталиране, поддържане и администриране на операционни системи и приложен софтуер по конкретно задание	
	Създаване на приложен софтуер по конкретно задание.	- създаване, тестване и настройване на програмно осигуряване, база от данни по конкретно задание	
	Създаване на вграден управляващ софтуер по конкретно задание.	- създаване, тестване и настройване на вграден управляващ софтуер (данни по конкретно задание)	
	Работа с интернет технологии и Web програмиране по конкретно задание.	- създаване, тестване и настройване на Web приложения по конкретно задание	
2.	Нормативни изисквания за организация на работното място.	- подготовка на компоненти, инструменти, апаратура и материали почистване и поддържане на работното място	3

3.	Здравословно и безопасно упражняване на професията.	- знания и умения за безопасна работа на работното място и при клиента - умение за адекватно реагиране в критични ситуации в рамките на компетенциите си - умение за оказване на първа помощ на пострадали при авария (при токов удар, пожар, наранявания и др.)	3
4.	Професионално-личностни качества.	- комуникативни умения - чувство на отговорност към извършената работа - трудова и технологична дисциплина - прецизност, сръчност, гъвкавост, мобилност и инициативност	6

Общ брой точки: 60

V. СИСТЕМА ЗА ОЦЕНЯВАНЕ

Максималният брой точки за всяка изпитна тема или за всяко изпитно задание е 60. Неправилният отговор се оценява с 0 точки. Непълният отговор се оценява с част от точките за верен и пълен отговор.

Преминаването от точки в цифрова оценка съгласно чл. 7, ал. 4 от Наредба №3 от 2003 г. за системата на оценяване се извършва по следната формула:

Цифрова оценка = общия брой точки от всички критерии : 10

Получената цифрова оценка се изчислява с точност до 0,01.

Оценяването на писмените работи от държавния изпит по теория е в съответствие с чл. 46 от Наредба № 3 от 2003 г. за системата на оценяване.

Изпълнението на практическото задание от държавния изпит по практика се оценява в съответствие с чл. 48 от Наредба № 3 от 2003 г. за системата на оценяване.

VI. ПРЕПОРЪЧИТЕЛНА ЛИТЕРАТУРА

1. **Войкова, А.** Икономика. Мартилен, С., 2003.
2. **Дончев, К.** и др. Охрана на труда и противопожарна охрана. Модул-96-ЕООД, С., 1996.
3. **Иванова, Л.** и колектив. Информатика за 9 клас. Програмиране на C++. Пловдив, Коала прес, 2006.
4. **Иванова, Л.** и колектив. Информатика за 9. Решения на задачите от Програмиране на C++. Пловдив, Коала прес, 2007.
5. **Колектив,** Microsoft Access 2003 Step by Step. СофтПрес, 2004.
6. **Миленкова, А.** Здравословни и безопасни условия на труд - Учебно помагало за задължителна професионална подготовка. С., Нови знания, 2005.
7. **Наков, Св.** Въведение в програмирането с Java. В. Търново., Faber, 2009.
<http://www.introprogramming.info/>
8. **Пенева, Ю.** Бази от данни. Регалия 6, 2005.
9. **Райкерсторфер, Г., К.** Вердених. Икономическа информатика. Тилиа, 1997.
10. **Стоянов, Б.** Ръководство за упражнения по програмиране на Асемблер първа част, Шумен, Химера, 2011. shu.bg/tadmin/upload/storage/179.pdf
11. **Тодорова, М.** Програмиране на C++, I и II част. София, 2002.
12. **Тодорова, М.** Обектно-ориентирано програмиране на базата на езика C++, Сиела, 2011.

13. **Тодорова**, М. Структури от данни и програмиране на C++, Сиела, 2011.
14. **Шилдт**, Хърбърт. JAVA. Ръководство на програмиста. Издателство Софт Прес, 2001.
15. **Cronan**, John Microsoft Office Access 2010 QuickSteps, McGraw-hill education – europe.
16. **Margolis**, Michael. Arduino Cookbook, O'Reilly Media, 2011, ISBN: 0596802471, 978-0596802479.
17. **McRoberts**, Michael. Beginning Arduino, Apress, 2010, ISBN:1430232404, 978-1430232407.

Електронни информационни източници:

<http://w3schools.com/> Сайт за обучение по уеб технологии

<http://www.cplusplus.com>

<http://www.nakov.com/blog/2012/04/13/bezplaten-kurs-programirane-za-mikrokontroleri-s-arduino-23-25-a...> – Безплатен курс по програмиране за микроконтролери Arduino

http://www.robotev.com/product_info.php?cPath=1_21&products_id=222 – Ардуино комплект за начинаещи

<http://vschool.info/programirane/>

VII. АВТОРСКИ КОЛЕКТИВ

1. **Мария Младенова** – СПГЕ „Джон Атанасов”, гр. София
2. инж. Красимир Дойчев – ПГ по МТЕ „М. В. Ломоносов”, гр. Добрич
3. инж. Валерия Василева – ПГ по телекомуникации, гр. София

VIII. ПРИЛОЖЕНИЯ

а) Примерен изпитен билет за държавния изпит по теория на професията и специалността

(пълно наименование на училището/обучаващата институция)

ДЪРЖАВЕН ИЗПИТ ПО ТЕОРИЯ НА ПРОФЕСИЯТА И СПЕЦИАЛНОСТТА ЗА ПРИДОБИВАНЕ НА ТРЕТА СТЕПЕН НА ПРОФЕСИОНАЛНА КВАЛИФИКАЦИЯ

по професия код 481020 „Системен програмист“
специалност код 4810201 „Системно програмиране“

Изпитен билет № 3

Изпитна тема: Основни оператори за управление на изчислителния процес в програмирането на C++

План-тезис: Основни оператори за управление на изчислителния процес в програмирането на C++ – оператор за присвояване, празен оператор, блок, условни оператори, оператори за цикъл. Необходима минимална конфигурация за инсталиране на езика за програмиране C++. Полагане на допълнителен труд по трудов договор. Правила за безопасност на работното място.

Приложни задачи:

Задача 1.

В зададения код на езика C++ открийте, документируйте и поправете грешките. Напишете кода на друг език за програмиране (езикът за програмиране е от изучаваните Pascal, Visual Basic, JavaScript или Java).

```
Include <iostream.h> ;
Int Mein ();
{
cout<<"n= ";
cin<<n
int n;
if (!cin)
{cout<<"Error. Bad input! \n";
Return 1;
}
If (n<=0){
Cout>>"Incorrect input! /n"; }
Return 1;
Int fact=1;
For (i=1, i<=n, i++)
Fact=fact*I;
Cout<<n<<"! = "<<fact<<"\n";
Return 0;
```

Задача 2.

Да се напише програма, която по дадени x - реално и n - естествено число пресмята сумата. Въведените стойности и резултатите запишете във файл.

$$S = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \dots + \frac{x^n}{n!}$$

Дидактически материали: справочни материали за синтаксиса на типовете данни и операторите на изучаваните програмни езици.

Председател на изпитната комисия: *Име фамилия*

.....
(подпис)

Директор на (наименование на училището/ : *Име фамилия*
обучаващата институция)

.....
(подпис) (печат)

Критерии и показатели за формиране на оценката на Изпитен билет № 3

№ по ред	Критерии за оценяване	Максимален брой точки
1.	Дефинира и обяснява основни оператори за управление на изчислителния процес в програмирането на посочения език	24
1.1.	Дефинира оператор за присвояване, съставни оператори за присвояване, празен оператор, блок.	6
1.2.	Дефинира условни оператори if, if-else, if-else if-else, switch-case.	9
1.3.	Дефинира оператори за цикъл for, while, do-while.	9
2.	Дефинира необходима минимална конфигурация за инсталиране на езика за програмиране	3
2.1.	Дефинира необходима минимална конфигурация за инсталиране на езика за програмиране C++ - хардуер и софтуер.	3
3.	Решава приложни задачи	
	Приложна задача 1	12
3.1.	Умее да разчита програми, написани на език за програмиране. Открива, документира и коригира синтактични и логически грешки в предоставения програмен код.	6
3.2.	Разбира алгоритъма на предоставена програма и пренаписва програмата на друг изучаван програмен език.	6
	Приложна задача 2	12
	Разработва програма на изучаван език за програмиране с използване на основни алгоритми и структури от данни.	
3.3.	Умее да създаде основната структура на програмата и да дефинира променливи и константи.	3
3.4.	Умее да въвежда и извежда данни.	3
3.5.	Владее основните алгоритми и операторите за управление на изчислителния процес и умее да ги прилага за обработка на данните.	6
4.	Теоретични знания в областта на трудовоправните закони	3
4.1.	Описва особеностите на полагане на допълнителен труд по трудов договор.	3
5.	Теоретични знания за безопасно упражняване на професията	3
5.1.	Изброява и обяснява правилата за безопасна работа на работното място.	3
6.	Основна професионална терминология в комуникацията на чужд език	3
6.1.	Използва и знае правописа и смисъла на основни понятия от професионалната терминология в комуникацията на чужд език.	3
	Общ брой точки:	60

Цифрова оценка = Получен общ брой точки : 10

Председател на изпитната комисия: *Име фамилия*

.....
(подпис)

Директор на (наименование на училището/ : *Име фамилия*
обучаващата институция)

.....
(подпис) (печат)

б) Примерно индивидуално практическо задание

(пълно наименование на училището/обучаващата институция)

**ДЪРЖАВЕН ИЗПИТ ПО ПРАКТИКА НА ПРОФЕСИЯТА И СПЕЦИАЛНОСТТА
ЗА ПРИДОБИВАНЕ НА ТРЕТА СТЕПЕН НА ПРОФЕСИОНАЛНА КВАЛИФИКАЦИЯ**

по професия код 481020 „Системен програмист“
специалност код 4810201 „Системно програмиране“

Индивидуално практическо задание № 1

На ученика.....
(трите имена на ученика)

от XII^A клас

начална дата на изпита:..... начален час:.....

крайна дата на изпита:..... час на приключване на изпита:.....

I. Задание

Задача 1 На работната станция да се инсталира и конфигурира операционна система Windows XP и интегрирана среда за разработване Borland C++.

Задача 2 Имаме отвор с размери AxB и кутия с размери XxYxZ.

На избран от ученика език за програмиране да се напише програма, в която да има клас Vox, съдържащ метод, който проверява дали съществува завъртане на кутията, при което тя минава през отвора, и да върне резултат – минава, не минава.

Да се изведе резултатът за 5 кутии с произволни размери за един и същи размер отвор в примерен формат:

Отвор: 245 x 385

Кутия 485 x 369 x 120 минава

Кутия 569 x 458 x 100 не минава

II. Указания за изпълнение на практическото задание:

Задача 1

1. Да се провери хардуерът дали отговаря на минималните изисквания за инсталиране.
2. Да се извърши инсталиране на операционната система.
3. Да се извърши инсталиране на драйверите на видеокартата и мрежовата карта.
4. Да се настрои операционната система: регионални и езикови настройки; мрежови настройки; дата и време; настройки на мишката.
5. Да се инсталира интегрираната среда за разработване.

Задача 2

1. Да се създаде клас, който да съдържа:
 - 1.1. полета с размерите на кутията – от реален тип с двойна точност;
 - 1.2. конструктор по подразбиране;
 - 1.3. конструктор с 3 параметъра - размерите на кутията; да се извършва проверка за валидност на данните – дали размерите са по-големи от 0;
 - 1.4. set методи за задаване стойности на всяко поле; да се извършва проверка за валидност на данните – дали размерите са по-големи от 0;
 - 1.5. get методи за извличане стойността на всяко поле;
 - 1.6. метод за проверка дали съществува завъртане на кутията, при което тя минава през отвора; методът да връща резултат „истина/лъжа“ и да извежда на екран съобщение от вида: „Кутия XxYxZ минава (не минава)“.
2. Да се създаде масив от 5 обекта от класа Vox, които получават стойности чрез конструктора с 3 параметъра.
3. На поне един обект да се променят стойностите на размерите чрез set методите.
4. Да се декларират и въведат размерите на отвора – от реален тип с двойна точност.
5. За създадените обекти да се извърши проверка дали минават през отвора и се изведат съответните съобщения в указания формат.
6. Да се извърши тестване на програмата с различни стойности, включително 0 и отрицателни.

III. По поставените задачи ученикът трябва да представи следното:

1. Работеща операционна система и софтуер за разработване на програми на C++.
2. Файлове на напълно работеща програма, записани на предоставен носител.

IV. Критерии за оценяване на индивидуално практическо задание №1

№	КРИТЕРИИ	ПОКАЗАТЕЛИ	Макс. бр. т.
1.	Проверка и оценяване на професионалните компетенции в следните тематични области:		48
	Поддържане и тестване на КС, ПУ, мрежови хардуер и съответен софтуер.	1.1 Избор на компоненти, модули, хардуерни и софтуерни средства съобразно практическото задание	2
	Инсталиране, конфигуриране и поддържане на операционни системи и приложен софтуер.	1.2 Инсталиране, поддържане и администриране на операционни системи и приложен софтуер по конкретно задание	7
	Създаване на приложен софтуер по конкретно задание.	1.3 Създаване на клас, отговарящ на изискванията на заданието. Създаване полета на класа	5
		1.4 Създаване на конструктори на класа	6
		1.5 Създаване на set и get методи на класа	6
		1.6 Създаване на методи на класа за изчисление и връщане на резултати	6
		1.7 Създаване на обекти от класа и задаване на свойствата им	6
		1.8 Извикване на методите за обработка на данните и извеждане на резултатите	6
		1.9 Тестване на продукта. Откриване, анализиране и отстраняване на грешки и проблеми	4
2.	Нормативни изисквания за организация на работното място. 3 т.	2.1 Подготовка на компоненти, инструменти, апаратура и материали	2
		2.2 Почистване и поддържане на работното място	1
3.	Здравословно и безопасно упражняване на професията. 3 т.	3.1 Знания и умения за безопасна работа на работното място и при клиента	1
		3.2 Умение за адекватно реагиране в критични ситуации в рамките на компетенциите си	1
		3.3 Умение за оказване на първа помощ на пострадал при авария (при токов удар, пожар, наранявания и др.)	1
4.	Професионално-личностни качества. 6 т.	4.1 Комуникативни умения	1
		4.2 Чувство на отговорност към извършената работа	1
		4.3 Трудова и технологична дисциплина	2
		4.4 Прецизност, сръчност, гъвкавост, мобилност и инициативност	2
		ОБЩО	60

Цифрова оценка = Получен общ брой точки : 10

Оценка:

Ученик:..... (име, фамилия) (подпис)

Председател на изпитната комисия: Име фамилия (подпис)

Директор на (наименование на училището/ : Име фамилия (подпис) (печат)
обучаващата институция)