Указатели

Променлива , която съдържа адреса на друга променлива(или обект). Всяка променлива (или друг обект) имат адрес в оперативната памет, където се съхраняват техните стойности, указателите извършват действия само с адресите на клетките от паметта. Когато декларираме указател е необходимо да се определи към какъв тип данни сочи.  
Тип \* Име на указател ;  
    **Int \*p;**  
**Тип -** към какъв тип променлива ще сочи указателя (int,double, float и т.н.)  
и в зависимост от това колко байта от паметта ще заеме.  
**Звездичката \***( в литературата я наричат оператор за дереференция) се   
използва:  
**Операторът &** връща адреса на променливата стояща след него.

Задача 1

1. #include<iostream>
2. using namespace std;
3. int main()
4. {
5. double \*p, r;
6. r = 3.456;
7. p = & r; // p взема адреса на r
8. cout<<\*p<< endl; // извежда 3.456
9. cout<< p<< endl; // извежда адреса на r
10. cout<<& p<< endl; // извежда адреса на указателя p
11. cout<<++p<< endl; // извежда адреса на r увеличен с 8
12. cout<<--p<< endl; // извежда адреса на r
13. cout<<\*p; // извежда 3.456
14. return 0;
15. }

Задача 2

Получаване на адрес на променлива и извличане на адрес &

1. #include <iostream>
2. using namespace std;
3. int main(){
4. unsigned long int a=10;
5. int b=20;
6. short c=-5;
7. double d=-3.15;
8. cout<<"a= "<<a<<"\tAdresut na a= "<<&a<<endl;
9. cout<<"b= "<<b<<"\tAdresut na b= "<<&b<<endl;
10. cout<<"c= "<<c<<"\tAdresut na c= "<<&c<<endl;
11. cout<<"d= "<<d<<"\tAdresut na d= "<<&d<<endl;
12. }

За променливи от тип int и long int се заделят 4 байта.

За short се заделят 2 байта.

За double се заделят 8 байта.

Задача 3

Деклариране и присвояване на значение на указател

1. #include <iostream>
2. using namespace std;
3. int main(){
4. int a=5;
5. int b=7;
6. cout<<"&a= "<<&a<< " ";
7. cout<<"&b= "<<&b<<endl;
8. int \*cd;
9. cd=&a;
10. cout<<"cd= "<<cd<<" ";
11. cd=&b;
12. cout<<"cd= "<<cd<<endl;
13. return 0;
14. }

На ред 8 е деклариран указател cd.

На редове 9 и 11 указателят cd се насочва последователно към променливите а и b.

Задача 4

#include <iostream>

using namespace std;  
int main()  
{  
int a=5,\*p;  
p=&a;  
cout<< p<< endl;  
cout<< \*p<< endl;  
return 0;  
}  
  
int a=5, \*p; - декларираме променлива а=5 и указател сочещ към целочислена променлива;  
p=&a; - това означава, че в указателя р се съдържа адреса на променливата а (нещо от рода 0х22ff14– в шестнадесетична бройна система 0х означава шестн. бр. система ).  
cout<< p<< endl; - извежда на екрана адреса на променливата а ;  
cout< <\*p; - ще изведе на екрана 5 т.е. компютъра ще отиде на адреса, който се съдържа   
в указателя р и от него ще извлече стойността ( на променливата а, която е 5).

Задача 5

Присвояване на значение на променливата.

1. #include <iostream>
2. using namespace std;
3. int main(){
4. int myAge=26;
5. int \*pmyAge= &myAge;
6. cout<<"myAge= "<<myAge<<endl;
7. cout<<"\*pmyAge= "<<\*pmyAge<<endl;
8. cout<<"Prisvoqvane na 20 na myAge "<<endl;
9. myAge=20;
10. cout<<"myAge= "<<myAge<<endl;
11. cout<<"\*pmyAge= "<<\*pmyAge<<endl;
12. cout<<"Prisvoqvane na 55 na myAge "<<endl;
13. \*pmyAge=55;
14. cout<<"myAge= "<<myAge<<endl;
15. cout<<"\*pmyAge= "<<\*pmyAge<<endl;
16. return 0;
17. }

На ред 5 указателя pmyAge e инициализиран с адреса на myAge.

На ред 9 променяме значението на myAge директно, а на ред 13 индиректно.

Задача 6

Какво ще изведе кодът?

1. int a=10, b=12, c=30;
2. int \*p1=&a;
3. int \*p2=&b;
4. int \*p3=&c;
5. cout<<"p3 "<<p3<<endl;
6. cout<<"p1 "<<\*p1<<endl;
7. cout<<"p2 "<<\*p2<<endl;

Задача 7

Какво ще изведе кодът?

1. unsigned short myAge=20, yourAge=26;
2. unsigned short \*pAge=&myAge;
3. cout<<"myAge: "<<myAge<<"\t yourAge: "<<yourAge<<endl;
4. cout<<"myAge: "<<&myAge<<"\t yourAge: "<<&yourAge<<endl;
5. cout<<"pAge: "<<pAge<<"\t pAge: "<<&pAge<<endl;
6. cout<<"pAge: "<<pAge<<"\t pAge: "<<\*pAge<<endl;
7. pAge=&yourAge;
8. cout<<"pAge: "<<\*pAge<<"\t pAge: "<<pAge<<endl;

Задача 8

Променете чрез указател стойността на променливата a от 5 на 20.

#include <iostream>

using namespace std;  
int main()  
{  
int a=5,\*p;  
p=&a;  
cout<< p<< endl;  
cout<< \*p<< endl;  
return 0;  
}

Задача 9

Какво ще изведе кодът?

1. int \* myptr;
2. int number1;
3. int number2;
4. myptr = &number1;
5. \*myptr = 5;
6. myptr = &number2;
7. \*myptr = 3;
8. number1 += number2;
9. cout << "number1 = "<< number1 << endl;
10. cout << "number2 = " << number2 << endl;

а) number1 = 8   
number2 = 3

б) number1 = 7   
number2 = 2

в) number1 = 9   
number2 = 4

а) грешка при компилация

Задача 10

1. int n = 10;
2. int\* pn = &n;
3. cout << "n= " << n << " \*pn= " << \*pn << '\n';
4. cout << "&n= "<< &n << " pn= " << pn << '\n';
5. int m = \*pn;
6. \*pn = 20;
7. cout << "n= " << n << '\n';
8. return 0;

а) n= 0x6ffe30 \*pn= 10

&n= 10 pn= 0x6ffe30

n= 20

б) n= 10 \*pn= 10

&n= 0x6ffe30 pn= 0x6ffe30

n= 20

в) n= 0x6ffe30 \*pn= 0x6ffe30

&n= 10 pn= 10

n= 20

г) n= 10 \*pn= 0x6ffe30

&n= 0x6ffe30 pn= 10

n= 20

Задача 11

Поправете грешката и изведете низ „Popraveno“.

1. char\* str;
2. cin >> str;
3. while (\*str)
4. {cout << \*str << '\n';
5. str++;

Задача 12

Поправете грешката.

1. int a[] = {1, 2, 3, 4, 5, 6};
2. for (int i = 0; i <= 5; i++)
3. cout << a[i] << '\n';
4. for (int i = 0; i <= 5; i++)
5. {cout << \*a << '\n';
6. a++;
7. }