

### 3.1 Assignment 1

```
int a = 3;      //tạo biến int a
int* b = &a;    //tạo pointer b đến reference của a

cout << b << endl; //in ra địa chỉ của a
cout << *b << endl; //in ra 3
cout << &b << endl; //in ra địa chỉ của b

cout << a << endl; //in ra 3
cout << &a << endl; //in ra địa chỉ của a
```

### 3.2 Assignment 2

```
int x, z;          //tạo biến int x, z
float y;           //tạo biến float y
char ch, *chp;     //tạo biến char ch và pointer chp
int* ip1, *ip2;    //tạo pointer ip1 và ip2
float* fp;

x = 100;           //đặt x = 100
y = 20.0;          //đặt y = 20.0
z = 50;            //đặt z = 50
ch = 'Z';          //đặt ch = 'Z'

ip1 = &x;          //đặt ip1 là địa chỉ của x
ip2 = &z;          //đặt ip2 là địa chỉ của z
fp = &y;           //đặt fp là địa chỉ của y
chp = &ch;         //đặt chp là địa chỉ của ch
ip2 = ip1;         //đổi ip2 thành địa chỉ của x
ip1 = &z;          //đổi ip1 thành địa chỉ z
*ip1 = *ip2;       //đổi z thành x
*ip1 = 200;        //đổi z thành 200
*ip1 = *ip2 + 300; //đổi z thành 100 + 300 = 400
*fp = 1.2;         //đổi y thành 1.2

cout << x << endl; //in ra 100
cout << y << endl; //in ra 1.2
cout << z << endl; //in ra 400
cout << ip1 << endl; //in ra địa chỉ của z
cout << *ip1 << endl; //in ra 400
cout << &ip1 << endl; //in ra địa chỉ của ip1
cout << ip2 << endl; //in ra địa chỉ của x
cout << *ip2 << endl; //in ra 100
cout << &ip2 << endl; //in ra địa chỉ của ip2
cout << fp << endl; //in ra địa chỉ của y
cout << *fp << endl; //in ra 1.2
cout << &fp << endl; //in ra địa chỉ của fp
cout << chp << endl; //in ra địa chỉ của ch
cout << *chp << endl; //in ra Z
cout << &chp << endl; //in ra địa chỉ của chp
```

### 3.3 Assignment 3

```
int* a = new int; //tạo pointer a đến một số nguyên
int* b = new int; //tạo pointer b đến một số nguyên
*a = 2;           //pointer a dẫn đến số nguyên 2
b = a;            //pointer b dẫn đến pointer a
```

```

cout << *a << endl;           //in ra 2
cout << *b << endl;           //in ra 2
delete a;                     //xóa a
delete b;                     //xóa b, lỗi do sau khi xóa a, b là pointer đến a sẽ point đến NULL

```

không thể xóa

### 3.4 Assignment 4

```

int a = 3;                     //tạo biến int a = 3
int* p = &a;                  //tạo pointer p đến a
cout << *p << endl;           //in ra 3
p = new int(5);               //p point tới số nguyên 5
cout << *p << endl;           //in ra 5

```

### 3.5 Assignment 5

```

int v = 8, * r, * s;          //tạo v = 8, pointer r và s
int* p;                       //tạo pointer p
int q = 100;                  //tạo q = 100
p = &q;                       //pointer p dẫn tới q
r = p;                       //pointer r dẫn tới p dẫn tới q
*p = 20;                      //pointer p dẫn tới 20
p = new int;                  //pointer p dẫn tới số int mới trống
*r = 30;                      //q = 30
q = v;                       //q = v = 8
s = p;                       //pointer s dẫn tới p là số int trống
*s = 50;                      //p = 50
cout << *p << endl;           //in ra 50
cout << q << endl;            //in ra 8
cout << *r << endl;           //in ra 8
cout << v << endl;            //in ra 8
cout << *s << endl;           //in ra 50

```

### 3.6 Assignment 6

```

int* p, * q, v, nom[5];       //tạo pointer p, q, biến int v, array 5 số int nom
p = &v;                       //p point tới v
*v = 12;                      //v = 12
q = p;                       //q point tới p point tới v
nom[0] = *q;                  //nom[0] = giá trị của q là giá trị của v = 12
p = nom;                      //p point tới array nom
p++;                         //p = nom[1]
nom[2] = 12;                  //nom[2] = 12
*p = 13;                     //nom[1] = 13
*q = 10;                     //p = 13
v = 11;                      //v = 11
*(p + 3) = 16;                //nom[4] = 16
p = &nom[3];                  //p point tới nom[3]
*p = 10;                     //nom[3] = 10
p--;                          //p = nom[2] = 12
cout << *p << endl;           //in ra 12
cout << *q << endl;           //in ra 11
cout << v << endl;            //in ra 11
cout << nom[0] << " ";        //in ra 12
cout << nom[1] << " ";        //in ra 13
cout << nom[2] << " ";        //in ra 12
cout << nom[3] << " ";        //in ra 10
cout << nom[4] << endl;       //in ra 16

```

### 3.7 Assignment 7

Answer is D because the error in this case is uninitialized variable x used. The pointer x is created and is not point to any memory, therefore we can't assign a value to its reference.

### 3.8 Assignment 8

Answer is D

### 3.9 Assignment 9

Answer is D, the s++ is executed after the printf function is used to output s + 3. The pointer s will move to str[3] and then print out the rest of the string.

### 3.10 Assignment 10

Answer is B, the change function will change the second value of the array to the last value of the array + 5.

### 3.11 Assignment 11

Answer is B, the array contains 5 int so its size is 20, the pointer to array is an integer so its size is 4, integer have 4 bytes.

### 3.12 Assignment 12

Answer is D, the pointer is created point to string "%d\n". str++ two times move it to the location str[2], str-2 will makes str back to original str, therefore print out 300 and new line.

### 3.13 Assignment 13

Answer is a.

### 3.14 Assignment 14

Answer is a

### 3.15 Assignment 15

Answer is d

### 3.16 Assignment 16

Answer is c

### 3.17 Assignment 17

Answer is b

### 3.18 Assignment 18

Answer is d, the program output the address of integer b.

### 3.19 Assignment 19

Answer is a.

### 3.20 Assignment 20

Answer is a.

### 3.21 Assignment 21

Answer is D

### 3.22 Assignment 22

Answer is D

### 3.23 Assignment 23

Answer is B

### 3.24 Assignment 24

Answer is C

### 3.25 Assignment 25

Answer is B

### 3.26 Assignment 26

Answer is A

### 3.27 Assignment 27

Answer is A

### 3.28 Assignment 28

Answer is C

### 3.29 Assignment 29

Answer is C

### 3.30 Assignment 30

Answer is C