Name: Châu Tấn Kiệt

ID: 21127329

Class: 21CLC4

3.1 Assignment 1

What is printed out? Are there any problems (errors)?

```
int a = 3;
int *b = &a;

cout << b << endl;
cout << *b << endl;
cout << &b << endl;

cout << a << endl;
cout << &a << endl;
```

cout << b << endl; will print out the memory address of variable a, because the pointer b is holding the memory address of variable a.

cout << *b << endl; will print out 3, because the pointer b is pointing to the value inside the address of variable a, which is 3

cout << &b << endl; will print out the memory address of the pointer b

cout << a << end; will print out the value inside variable a, which is 3;

cout << &a << endl; will print out the memory address of variable a, similar to cout << b << endl;

3.3 Assignment 3

What is printed out? Are there any problems (errors)?


```
int *a = new int;
int *b = new int;
*a = 2;
b = a;
cout << *a << endl;
cout << *b << endl;
delete a;
delete b;
```

int *a = new int;

int *b = new int;

*a = 2;

b = a; → *b = *a;

cout << *a << endl;

cout << *b << endl;

delete a;

delete b;

After allocating the new variables new int on both of the pointers a and b, we'll have to sign on the value on the pointer *a with the new value of *b, with *b = *a, not b = a

The result is

2

2

3.4 Assignment 4

What is printed out? Are there any problems (errors)?

```
int a = 3;
int *p = &a;
cout << *p << endl;
p = new int(5);
cout << *p << endl;
```

int a = 3;

int* p = &a;

cout << *p << endl;

p = new int(5);

cout << *p << endl;

The results will be:

3

5

Because the first cout << *p << endl; points to the value inside variable a, which is 3, the second cout << *p << endl; was asking a memory cell of 5 bytes, which is also signed with a new value of 5.

After using the given memory cell, we'll have to return the memory cell, using delete p.

3.7 Assignment 7

1. Point out the compile time error in the program given below.

```
#include<stdio.h>

int main()
{
    int *x;
    *x=100;
    return 0;
}
```

A. Error: invalid assignment for x

B. Error: suspicious pointer conversion

C. No error

D. None of above

Uninitialized pointer x so we will have to dynamically allocate a single variable to the pointer x

Fixed code:

#include <stdio.h>

Int main() {

int *x = new int;

*x =100;

return 0;

}

So the answer is B.Invalid assignment for x

3.9 Assignment 9

16. What will be the output of the program ?

```c
#include<stdio.h>

int main()
{
    char str[] = "peace";
    char *s = str;
    printf("%s\n", s++ +3);
    return 0;
}
```

A. peace

B. eace

C. ace

D. ce

The answer is D. ce. There are 5 characters in the char str[] = “peace”.

And the printf commands to skip the first 3 characters(+3) amd print the rest of the characters (s++)

3.13 Assignment 13

2. The operator used for dereferencing or indirection is _____
a) *
b) &
c) ->
d) –>>

The answer is a)* because it is used to read the value stored inside the pointed address.

3.14 Assignment 14

3. Choose the right option
string* x, y;
a) x is a pointer to a string, y is a string
b) y is a pointer to a string, x is a string
c) both x and y are pointers to string types
d) none of the mentioned

The answer is a) because the pointer is grouped to a variable, but not the data type, so y is still a string

3.15 Assignment 15

4. Which one of the following is not a possible state for a pointer.
a) hold the address of the specific object
b) point one past the end of an object
c) zero
d) point to a tye

The answer is d) because a pointer can only be either a), b), c), but it cannot be d)

3.16 Assignment 16

5. Which of the following is illegal?

a) int *ip;

b) string s, *sp = 0;

c) int i; double* dp = &i;

d) int *pi = 0;

The answer is c) because a double value pointer cannot point to a value of int

3.17 Assignment 17

6. What will happen in this code?

```
1.    int a = 100, b = 200;
2.    int *p = &a, *q = &b;
3.    p = q;
```

a) b is assigned to a

b) p now points to b

c) a is assigned to b

d) q now points to a

The answer is b) p = q means that p will now points to the value inside the address of b

3.18 Assignment 18

7. What is the output of this program?

```
1.  #include <iostream>
2.  using namespace std;
3.  int main()
4.  {
5.      int a = 5, b = 10, c = 15;
6.      int *arr[ ] = {&a, &b, &c};
7.      cout << arr[1];
8.      return 0;
9.  }
```

a) 5
b) 10
c) 15
d) it will return some random number

The answer is d) because an array element with type int cannot be an address of a variable

3.19 Assignment 19

9. What is the output of this program?

```
1.  #include <iostream>
2.  using namespace std;
3.  int main()
4.  {
5.      char arr[20];
6.      int i;
7.      for(i = 0; i < 10; i++)
8.          *(arr + i) = 65 + i;
9.      *(arr + i) = '\0';
10.     cout << arr;
11.     return(0);
12. }
```

a) ABCDEFGHIJ
b) AAAAAAAAAA
c) JJJJJJJJ
d) None of the mentioned

The answer is a). At the value i = 0. It will be 65, and onwards it will print from A to J

3.20 Assignment 20

10. What is the output of this program?

```
1.  #include <iostream>
2.  using namespace std;
3.  int main()
4.  {
5.      char *ptr;
6.      char Str[] = "abcdefg";
7.      ptr = Str;
8.      ptr += 5;
9.      cout << ptr;
10.     return 0;
11. }
```

a) fg
b) cdef
c) defg
d) abcd

The answer is a) The pointer skips the first 5 strings abcde, and it points to string fg

3.21 Assignment 21

MCQ: A pointer can be initialized with

A. Null
B. Zero
C. Address of an object of same type
D. All of them

The answer is D

3.22 Assignment 22

MCQ: Which from following is not a correct way to pass a pointer to a function?

A. Non-constant pointer to non-constant data
B. A non-constant pointer to constant data
C. A constant pointer to non-constant data
D. All of them

The answer is D. All of them

3.23 Assignment 23

MCQ: A qualifier that enables programmers to inform compiler that value of a particular variable should not be modified?

A. ptr
B. const
C. stsrt
D. None of them

The answer is B. Const

3.24 Assignment 24

MCQ: Which operator returns address of unallocated blocks in memory?

A. The delete operator
B. The empty operator
C. The new operator
D. All of them

The ansswer iss B

3.25 Assignment 25

MCQ: Referencing a value through a pointer is called

A. Direct calling
B. Indirection
C. Pointer referencing
D. All of them

The ansswer is B, Indirection

3.26 Assignment 26

MCQ: Which unary operator is used for determining size of an array?

A. sizeof
B. size_array
C. s_array
D. size_ofarray

The answer is A. sizeof

3.27 Assignment 27

MCQ: What is a Pointer?

A. Pointer contains an address of a variable
B. It's an operator
C. It?s a function
D. None of them

The answer is A. It contains an address of a variable

3.28 Assignment 28

MCQ: There are how many values that can be used to initialize a pointer

A. 1
B. 2
C. 3
D. 4

The answer is C. There are 3 values: 0, NULL or an address.

3.29 Assignment 29

MCQ: A unary operator that returns address of its operands, are called

A. Pointer operator
B. Relationship operator
C. Address operator
D. Both A and B

The answer is C

3.33Assignment 33

4. What will be the output of the following program?

```
#include <iostream>
using namespace std;
int main()
{
    int arr[] = { 4, 5, 6, 7 };
    int* p = (arr + 1);
    cout << *arr + 10;
    return 0;
}
```

Options:

a. 12

b. 15

c. 14

d. error

The answer is C. Because we're adding 10 to the initial value of the array, so the answer is 14

3.35Assignment 35

1. **Question 1**

   What will be the output?

```
#include<stdio.h>

int main()
{
    int a[] = { 1, 2, 3, 4, 5} ;
    int *ptr;
    ptr = a;
    printf(" %d ", *( ptr + 1) );

    return 0;
}
```

The output will be 1 because *(ptr+1) prints out the initial value of the array, which is 1

3.43Assignment 43

Question 4 : What will be the output of following program?

```c
#include <stdio.h>
int main()
{
    int num = 10;
    printf("num = %d addresss of num = %u", num, &num);
    num++;
    printf("\n num = %d addresss of num = %u", num, &num);
    return 0;
}
```

OPTIONS:

a) Compilation error

b) num = 10 address of num = 2293436
num = 11 address of num = 2293438

c) num = 10 address of num = 2293436
num = 11 address of num = 2293440

d) num = 10 address of num = 2293436
num = 11 address of num = 2293436

The answer is D. Only the value is changed, not the address

3.44Assignment 44

Question 5 : What will be the output of following program?

```c
#include <stdio.h>
int main()
{
    int i = 25;
    int* j;
    int** k;
    j = &i;
    k = &j;
    printf("%u %u %u ", k, *k, **k);
    return 0;
}
```

OPTIONS:

a) address address value

b) address value value

c) address address address

d) compilation error

The answer is A. k = the address of j. *k = the address of i, **k points to the value inside the address of i.

3.45 Assignment 45

QUE.1 What would be printed from the following C++ program?

```
#include <iostream>
#include <stdlib.h>
using namespace std;
int main()
{
    float x = 5.999;
    float* y, *z;
    y = &x;
    z = y;
    cout << x << ", " << *(&x) << ", " << *y << ", " << *z << "\n";
    return 0;
}
```

a) 5.999, 5.999, 5.999, 5.999

b) 5.999, 5.9, 5.000, 5.900

c) Address of the elements

d) compilation error

The answer is A. X is the value of x, *(&x) points to the value inside the address of x, same with *y and *z.

3.38 Assignment 38

4. **Question 4**

What will be the output?

```
#include<stdio.h>

int main()
{
    int x = 20, *y, *z;

    // Assume address of x is 500 and
    // integer is 4 byte size
    y = &x;
    z = y;
    *y++;
    *z++;
    x++;
    printf("x = %d, y = %d, z = %d \n", x, y, z);
    return 0;
}
```

The output will be 21, 500, 500. It prints out the value of x after adding one, and the address of x for y and z.

3.41 Assignment 41

Question 2 : What will be the output of following program?

```c
#include <stdio.h>
int main()
{
    int* ptr;
    *ptr = 5;
    printf("%d", *ptr);
    return 0;
}
```

OPTIONS:

a) compilation error

b) Runtime error

c) 5

d) linker error

The answer is a) The pointer has no address to point to, so it will cause compilation error

3.42 Assignment 42

Question 3 : What will be the output of following program?

```c
#include <stdio.h>
int main()
{
    int a = 36;
    int* ptr;
    ptr = &a;
    printf("%u %u", *&ptr, &*ptr);
    return 0;
}
```

OPTIONS:

a) Address Value

b) Value Address

c) Address Address

d) Compilation error

The answer is c) *&ptr ponts to the address of ptr, &*ptr is the address of *ptr.