**Name: Abdul Ghaffar Kalhoro**

**Registration no: 194699**

**Class: BSCS- 6C 3rd semester**

**Lab\_1**

**Task 1:**

**Source file:**

#include<iostream>

using namespace std;

//main function

int main(void)

{

//declaring pointer

int\* salary;

salary = new int[20];

int i;

for (i = 0; i<20; ++i)

{

cout << "Enter Salary: ";

cin >> salary[i];

}

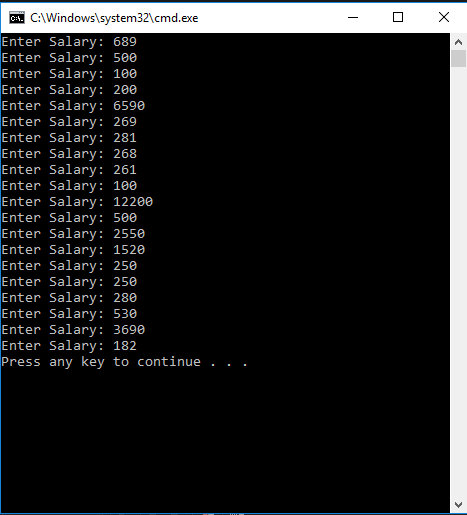
for (i = 0; i<20; ++i)

salary[i] = salary[i] + salary[i] / (i + 1);

return 0;

} //end main function.

**OUTPUT**

****

**Task 2:**

**Source file:**

**#include<iostream>**

**using namespace std;**

**//main function**

**//declaring and initializing the function**

**void analyze\_pointer(int \*ptr){**

**cout << "address = ";**

**//printing the address of ptr**

**cout << ptr << endl;**

**cout << "value = ";**

**//printing the value of ptr**

**cout << \*ptr;**

**}**

**int \* heap(){**

**int\* pointer = new int;**

**\*pointer = 29;**

**return pointer;**

**}**

**int \* stack(){**

**int\* STACK;**

**int value = 53;**

**return STACK = &value;**

**}**

**int main(void)**

**{**

**//part 1 of task 2**

**cout << "part 1"<< endl;**

**int\*number;**

**int value;**

**value = 20;**

**number = &value;**

**//calling the function analyze\_pointer**

**analyze\_pointer(number);**

**//part2 of task 2**

**int \*x = new int;**

**\*x = 90;**

**cout << "\npart 2" << endl;**

**//calling the function analyze\_pointer**

**analyze\_pointer(x);**

**cout << "\n\n heap function: ";**

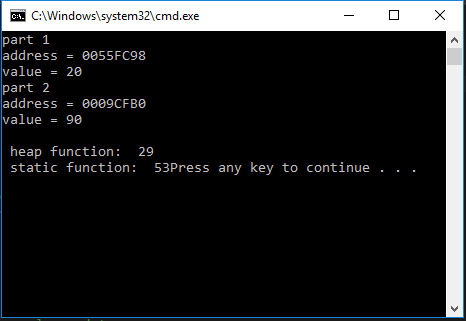
**cout << \*heap();**

**cout << "\n static function: ";**

**cout << \*stack();**

**} //end main function**

**OUTPUT**

****

Task 2:

Question #1. What happens?

Answer: In part 1 the reference of pointer as written while it is not stored but saved in a memory which is called static memory allocation but in 2nd part we have created the pointer using heap in which the 'new' keyword is used in which the space is allocated directly in computer and is called as the heap memory allocation.