**Report of Programming Practicum X**

Nurrizky Imani

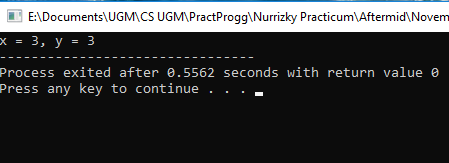
18/429294/PA/18685

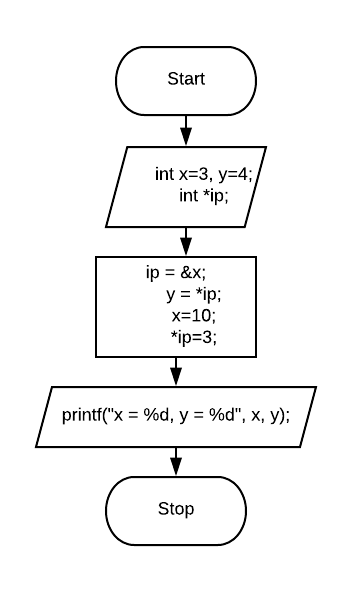
**Exercise 1:**

|  |
| --- |
| #include <iostream> |
|  | #include<stdio.h> |
|  | using namespace std; |
|  |  |
|  | int main () { |
|  | int x=3, y=4; |
|  | int\*ip; |
|  | ip = &x; |
|  | y = \*ip; |
|  | x=10; |
|  | \*ip=3; |
|  | printf("x = %d, y = %d", x, y); |
|  | return 0; |
|  | } |

**Explanation :**

In this program, we would like to assign two integer: x= 3, y=4. Then we initialize int \*ip in which refers to storing memory address. We assign ip as &x which mean “ip is filled with the address of x”. Then we assing y as the data not the address of the ip. X is assigned with 10 while \*ip is 3 which is going to be assign in y as well. As we print the x and y. The result would be the data not the address itself. So clearly that y is referring to \*ip not the ip only which refers to the address of x.





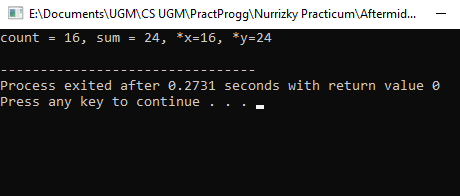
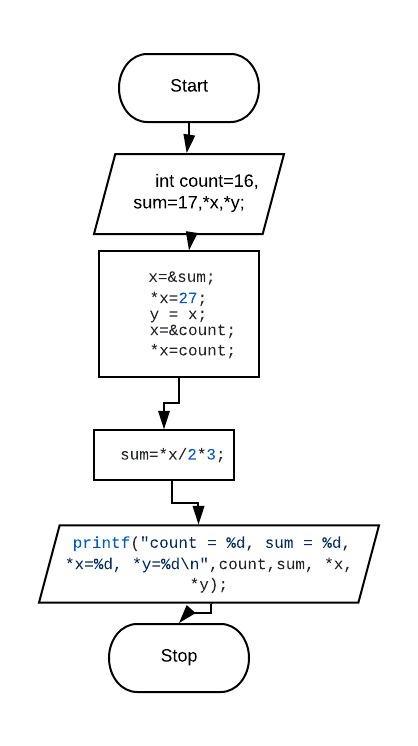
**Exercise 2 :**

|  |
| --- |
| #include <iostream> |
|  | #include<stdio.h> |
|  | using namespace std; |
|  |  |
|  | int main () { |
|  | int count=16, sum=17,\*x,\*y; |
|  | x=&sum; |
|  | \*x=27; |
|  | y = x; |
|  | x=&count; |
|  | \*x=count; |
|  | sum=\*x/2\*3; |
|  | printf("count = %d, sum = %d, \*x=%d, \*y=%d\n",count,sum, \*x, \*y); |
|  | return 0; |
|  | } |

**Explanation :**

In this program, first we would like to initialize several variable integer : count = 16, sum =17, \*x, \*y. Notice that x and y assign with star for announce that the are could be assign in pointer. Then x is assign with the address of &sum while \*x will assign with variable of sum = 27. Then we assign y to be x which is the address of &sum when \*x is equal with 16. There is operation as sum is equal of 27/2\*3.

We then print all of the variable of count, sum \*x, \*y. The count is equal 16, sum is equal to 24 as coming from the operation above while \*x equal with 16 and y equal to 24 with the reasons that y\* refers to the variable that had assign in x in which follow the change in the operation sum above. Looking more further, we can see that y is assign as x as it is the address. Therefore the \*y is the reversion of y itself which is the variable instead of the address.

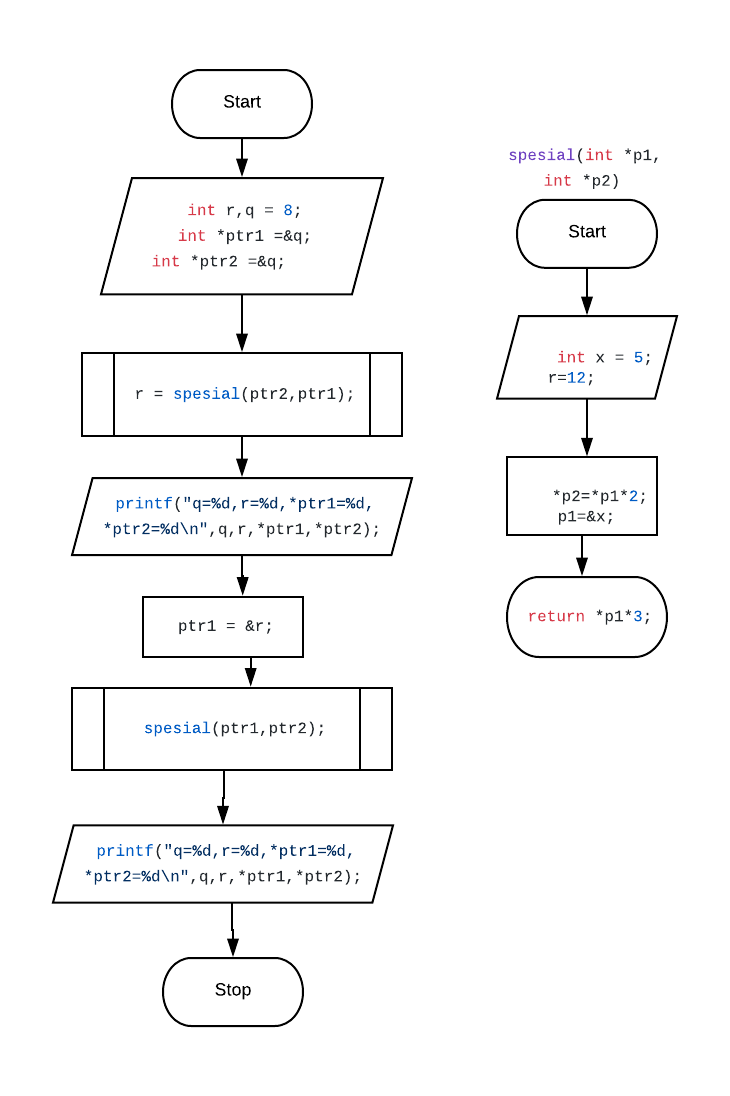
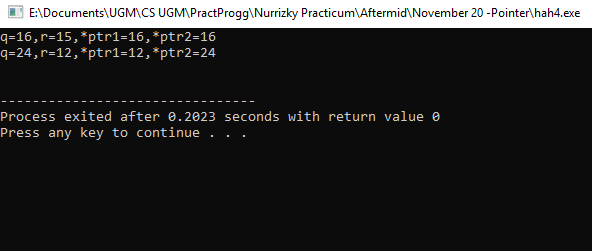


**Exercise 3 :**

|  |
| --- |
| #include <iostream> |
|  | #include <stdio.h> |
|  | using namespace std; |
|  |  |
|  | int r,q = 8; |
|  | int spesial(int\*,int\*); |
|  | main() { |
|  | int \*ptr1 =&q; |
|  | int \*ptr2 =&q; |
|  | r = spesial(ptr2,ptr1); |
|  | printf("q=%d,r=%d,\*ptr1=%d,\*ptr2=%d\n",q,r,\*ptr1,\*ptr2); |
|  | ptr1 = &r; |
|  | spesial(ptr1,ptr2); |
|  | printf("q=%d,r=%d,\*ptr1=%d,\*ptr2=%d\n",q,r,\*ptr1,\*ptr2); |
|  | } |
|  | int spesial(int \*p1, int \*p2) |
|  | { |
|  | int x = 5; |
|  | r=12; |
|  | \*p2=\*p1\*2; |
|  | p1=&x; |
|  | return \*p1\*3; |
|  | } |

**Explanation :**

In this program, we assign integer r, q = 6 and function with input of two integer. we create function special with two input \*p1 and \*p2. Inside the function we initialize x=5 and r=12. We then do operation using variable of \*p \* 2 assign it in \*p2. While p1 is assign as the address of &x. We then return value of operation of \*p1\*3 which is variable. We then inputing into the main function as assign in r. Thus, r is the result of \*p1\*3 which is variable. The result for the first print would be : q=16, r=15, \*ptr1=16, \*ptr2=16 while second print would be q=24, r=12, \*ptr1=12, \*ptr2=24 after doing the second special function.



**Exercise 4 :**

|  |
| --- |
| #include <iostream> |
|  | #include <stdio.h> |
|  | using namespace std; |
|  |  |
|  | main () { |
|  | int var\_x=273; |
|  | int \*ptr1; |
|  | int \*\*ptr2; |
|  | ptr1= &var\_x; |
|  | ptr2= &ptr1; |
|  | printf("Nilai var\_x=\*ptr1=%d\n",\*ptr1); |
|  | printf("Nilai var\_x=\*\*ptr2=%d\n\n",\*\*ptr2); |
|  | printf("ptr1 = &var\_x=%p\n",\*ptr1); |
|  | printf("ptr2 = &ptr1=%p\n",\*ptr1); |
|  | printf(" &ptr2=%p\n",&ptr2); |
|  | } |

**Explanation :**

In this program, we would like to initialize several integer var\_x = 273, \*ptr1 and \*\*ptr2.Then we will assign ptr1 as the address of var\_x and the ptr 2 will be the address of ptr1. Thus when we print the the \*ptr1 and \*\*ptr2 it is the opposite that we already assign, which is not the address but the variable itself 273. However, when we print the \*ptr1 next it will output the address as we use the %p in the print function indicate that the program want the address of it.

