

LAPORAN PRAKTIKUM
ALGORITMA PEMROGRAMAN
“POST TEST 7: Sorting dan Searching”

Diajukan untuk memenuhi salah satu praktikum Mata Kuliah Algoritma Pemrograman yang
di ampu oleh:

Dr. Ardiansyah S.T., M.Cs



Disusun Oleh:

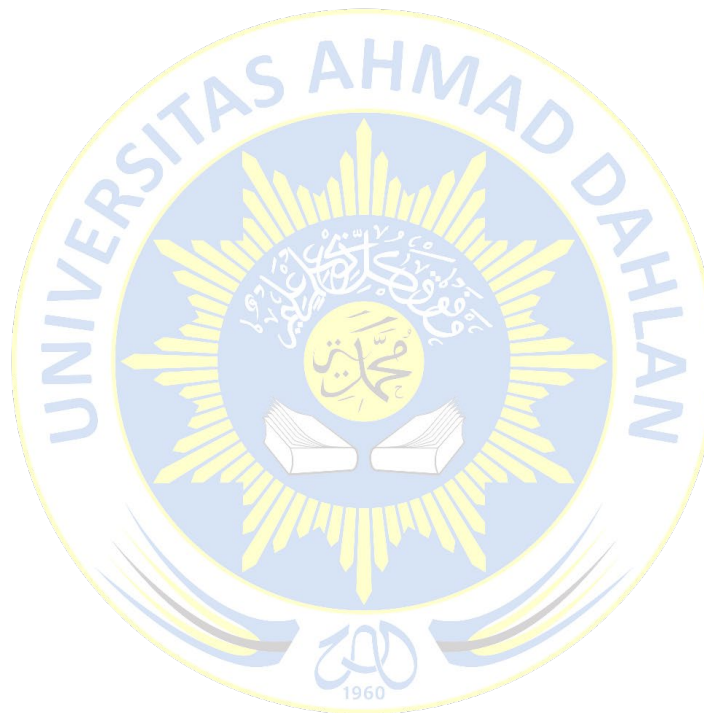
Mohammad Farid Hendianto 2200018401

A / Jumat 13.30 Lab. Jaringan

PROGRAM STUDI INFORMATIKA
UNIVERSITAS AHMAD DAHLAN
FAKULTAS TEKNOLOGI INDUSTRI
TAHUN 2023

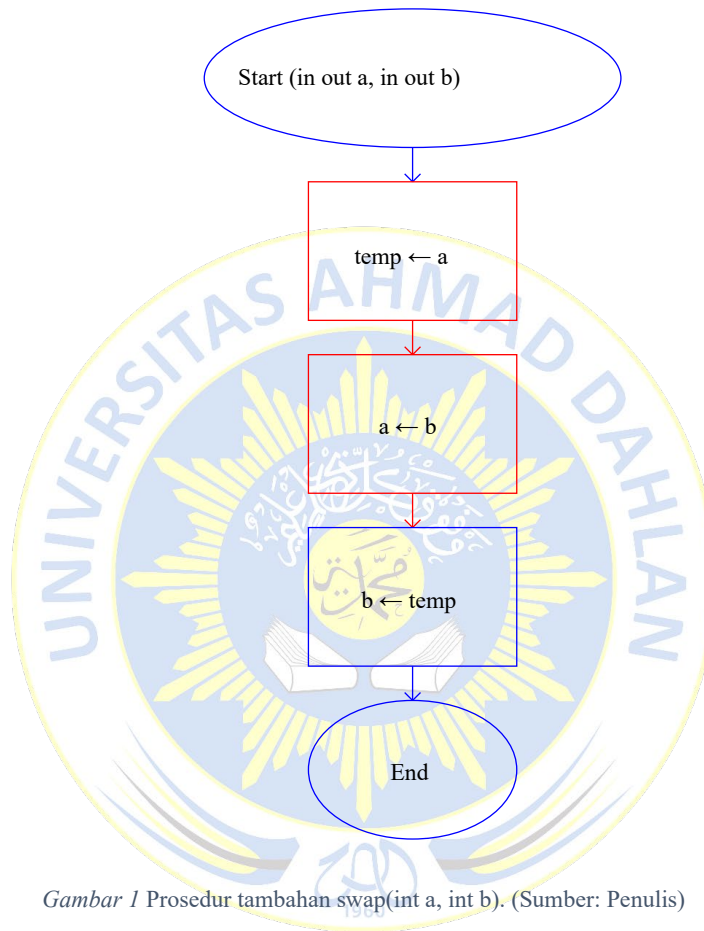
DAFTAR SOAL

1. Buat lah flowchart untuk mengurutkan deret bilangan **9 2 1 4 11 10 18 6** dengan bubble sort dan tampilkan tiap langkah sortingnya secara manual..... 3
2. Seperti nomor 1, gunakan subprogram dalam flowchart untuk mengurutkan deret bilangan **9 2 1 4 11 10 18 6** dengan bubble sort dan tampilkan tiap langkah sortingnya secara manual..... 11
3. Konversikan hasil dari flowchart nomor 1 dan 2 menjadi program C++..... 21



1. Buat lah flowchart untuk mengurutkan deret bilangan **9 2 1 4 11 10 18 6** dengan bubble sort dan tampilkan tiap langkah sortingnya secara manual.

Berikut adalah flowchart:



Gambar 1 Prosedur tambahan swap(int a, int b). (Sumber: Penulis)



Algoritma yang digunakan dalam flowchart ini adalah Bubble Sort. Bubble Sort adalah algoritma pengurutan yang sederhana dan bekerja dengan cara membandingkan setiap pasangan elemen yang berdekatan dalam array dan menukarnya jika mereka tidak berurutan.

Flowchart ini terdiri dari sebuah prosedur bantuan bernama swap. Prosedur swap digunakan untuk menukar nilai dua variabel yang diberikan sebagai input.

Flowchart memiliki beberapa variabel dan array sebagai berikut:

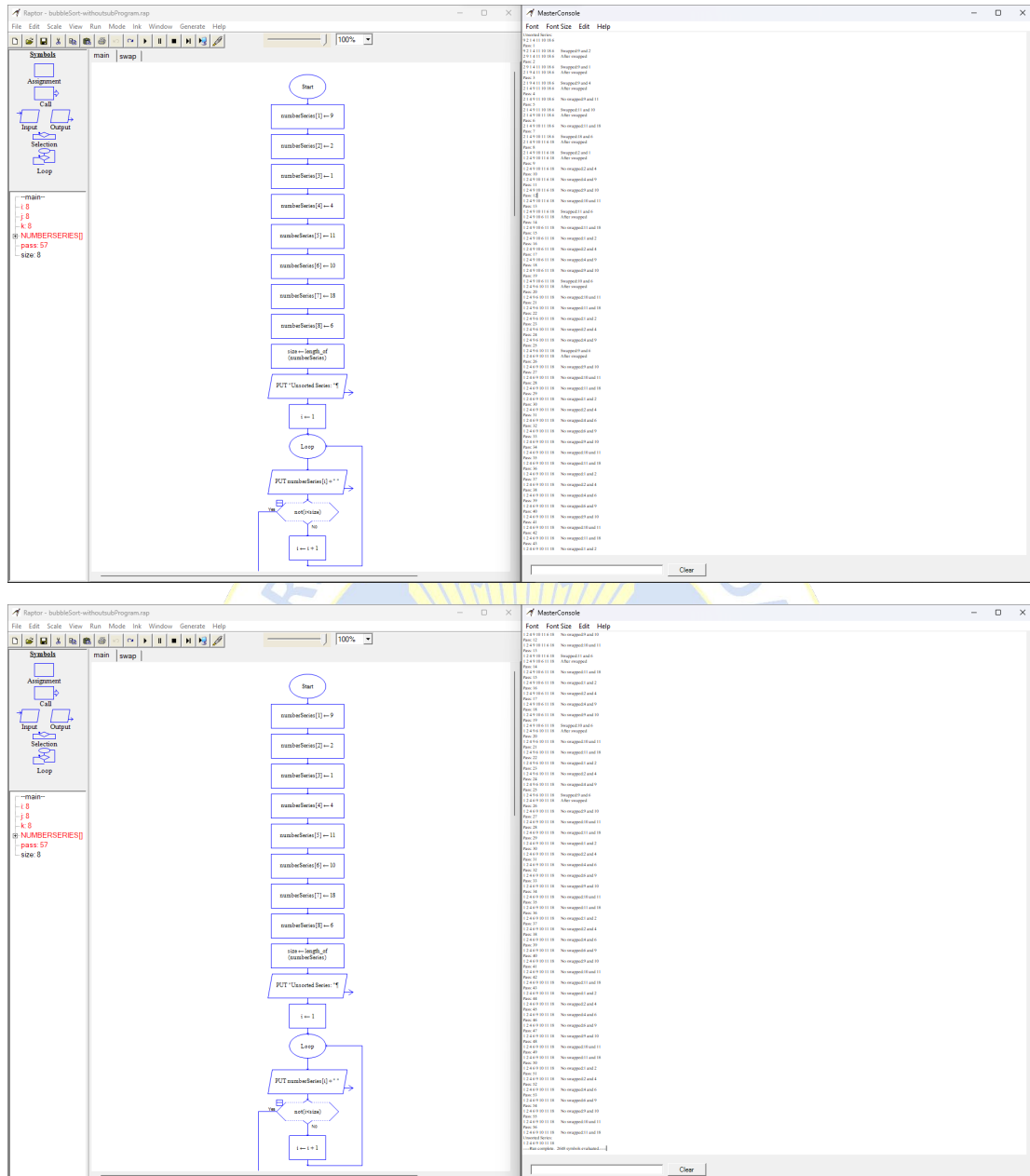
- numberSeries: Array 1D yang berisi angka-angka yang akan diurutkan.
- size: Variabel yang menyimpan panjang dari array numberSeries.
- i, j, k, dan pass: Variabel penghitung untuk iterasi dalam loop.

Alur flowchart tersebut adalah:

- 1) Mengisi array numberSeries dengan angka-angka yang akan diurutkan.
- 2) Menampilkan angka-angka dalam array numberSeries sebelum diurutkan.
- 3) Menginisialisasi variabel pass, i, dan j dengan nilai 1.
- 4) Melakukan loop luar dengan variabel i sebagai penghitung. Loop ini akan berjalan sebanyak size - 1 kali.
- 5) Melakukan loop dalam dengan variabel j sebagai penghitung. Loop ini akan berjalan sebanyak size - 1 kali.
- 6) Menampilkan nilai pass saat ini.
- 7) Membandingkan elemen numberSeries(j) dengan elemen numberSeries(j+1). Jika elemen numberSeries(j) lebih besar dari elemen numberSeries(j+1), maka: a. Menampilkan elemen yang akan ditukar. b. Menukar elemen numberSeries(j) dengan elemen numberSeries(j+1) menggunakan prosedur swap. c. Menampilkan elemen setelah ditukar.
- 8) Jika elemen numberSeries(j) tidak lebih besar dari elemen numberSeries(j+1), maka menampilkan bahwa tidak ada penukaran elemen.
- 9) Mengincrement nilai j dan pass.
- 10) Mengulangi langkah 6-9 hingga loop dalam selesai.
- 11) Mengincrement nilai i.
- 12) Mengulangi langkah 5-11 hingga loop luar selesai.
- 13) Menampilkan angka-angka dalam array numberSeries setelah diurutkan.

Dengan algoritma ini, array numberSeries akan diurutkan secara ascending menggunakan metode Bubble Sort.

Berikut adalah outputnya:



Gambar 3 Output Raptor dengan Flowchart tanpa subProgram (kecuali swap). (Sumber: Penulis)

Berikut logs output berbentuk teks:

Unsorted Series:

9 2 1 4 11 10 18 6

Pass: 1

9 2 1 4 11 10 18 6 Swapped:9 and 2

2 9 1 4 11 10 18 6 After swapped

Pass: 2

2 9 1 4 11 10 18 6 Swapped:9 and 1

2 1 9 4 11 10 18 6 After swapped

Pass: 3

2 1 9 4 11 10 18 6 Swapped:9 and 4

2 1 4 9 11 10 18 6 After swapped

Pass: 4

2 1 4 9 11 10 18 6 No swapped:9 and 11

Pass: 5

2 1 4 9 11 10 18 6 Swapped:11 and 10

2 1 4 9 10 11 18 6 After swapped

Pass: 6

2 1 4 9 10 11 18 6 No swapped:11 and 18

Pass: 7

2 1 4 9 10 11 18 6 Swapped:18 and 6

2 1 4 9 10 11 6 18 After swapped

Pass: 8

2 1 4 9 10 11 6 18 Swapped:2 and 1

1 2 4 9 10 11 6 18 After swapped

Pass: 9

1 2 4 9 10 11 6 18 No swapped:2 and 4

Pass: 10

1 2 4 9 10 11 6 18 No swapped:4 and 9

Pass: 11



1 2 4 9 10 11 6 18 No swapped:9 and 10

Pass: 12

1 2 4 9 10 11 6 18 No swapped:10 and 11

Pass: 13

1 2 4 9 10 11 6 18 Swapped:11 and 6

1 2 4 9 10 6 11 18 After swapped

Pass: 14

1 2 4 9 10 6 11 18 No swapped:11 and 18

Pass: 15

1 2 4 9 10 6 11 18 No swapped:1 and 2

Pass: 16

1 2 4 9 10 6 11 18 No swapped:2 and 4

Pass: 17

1 2 4 9 10 6 11 18 No swapped:4 and 9

Pass: 18

1 2 4 9 10 6 11 18 No swapped:9 and 10

Pass: 19

1 2 4 9 10 6 11 18 Swapped:10 and 6

1 2 4 9 6 10 11 18 After swapped

Pass: 20

1 2 4 9 6 10 11 18 No swapped:10 and 11

Pass: 21

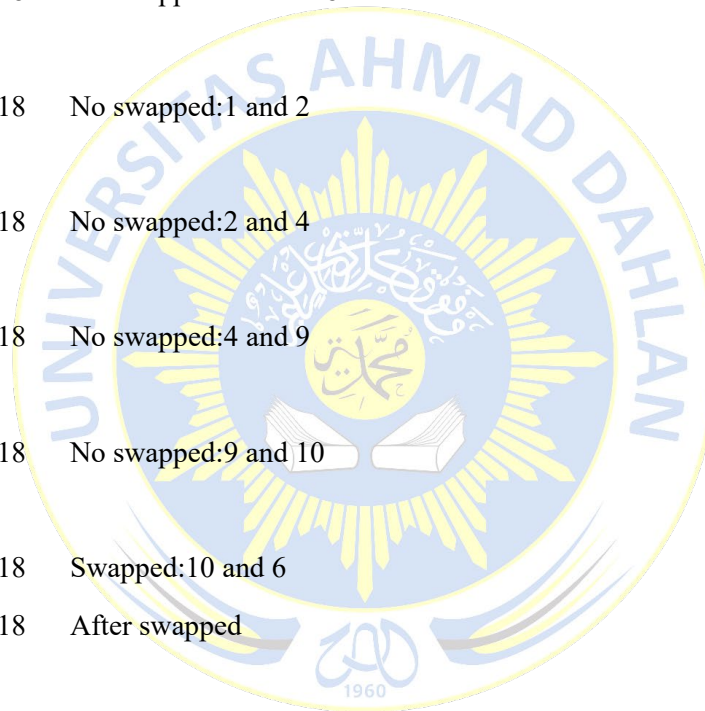
1 2 4 9 6 10 11 18 No swapped:11 and 18

Pass: 22

1 2 4 9 6 10 11 18 No swapped:1 and 2

Pass: 23

1 2 4 9 6 10 11 18 No swapped:2 and 4



Pass: 24

1 2 4 9 6 10 11 18 No swapped:4 and 9

Pass: 25

1 2 4 9 6 10 11 18 Swapped:9 and 6

1 2 4 6 9 10 11 18 After swapped

Pass: 26

1 2 4 6 9 10 11 18 No swapped:9 and 10

Pass: 27

1 2 4 6 9 10 11 18 No swapped:10 and 11

Pass: 28

1 2 4 6 9 10 11 18 No swapped:11 and 18

Pass: 29

1 2 4 6 9 10 11 18 No swapped:1 and 2

Pass: 30

1 2 4 6 9 10 11 18 No swapped:2 and 4

Pass: 31

1 2 4 6 9 10 11 18 No swapped:4 and 6

Pass: 32

1 2 4 6 9 10 11 18 No swapped:6 and 9

Pass: 33

1 2 4 6 9 10 11 18 No swapped:9 and 10

Pass: 34

1 2 4 6 9 10 11 18 No swapped:10 and 11

Pass: 35

1 2 4 6 9 10 11 18 No swapped:11 and 18

Pass: 36

1 2 4 6 9 10 11 18 No swapped:1 and 2



Pass: 37

1 2 4 6 9 10 11 18 No swapped:2 and 4

Pass: 38

1 2 4 6 9 10 11 18 No swapped:4 and 6

Pass: 39

1 2 4 6 9 10 11 18 No swapped:6 and 9

Pass: 40

1 2 4 6 9 10 11 18 No swapped:9 and 10

Pass: 41

1 2 4 6 9 10 11 18 No swapped:10 and 11

Pass: 42

1 2 4 6 9 10 11 18 No swapped:11 and 18

Pass: 43

1 2 4 6 9 10 11 18 No swapped:1 and 2

Pass: 44

1 2 4 6 9 10 11 18 No swapped:2 and 4

Pass: 45

1 2 4 6 9 10 11 18 No swapped:4 and 6

Pass: 46

1 2 4 6 9 10 11 18 No swapped:6 and 9

Pass: 47

1 2 4 6 9 10 11 18 No swapped:9 and 10

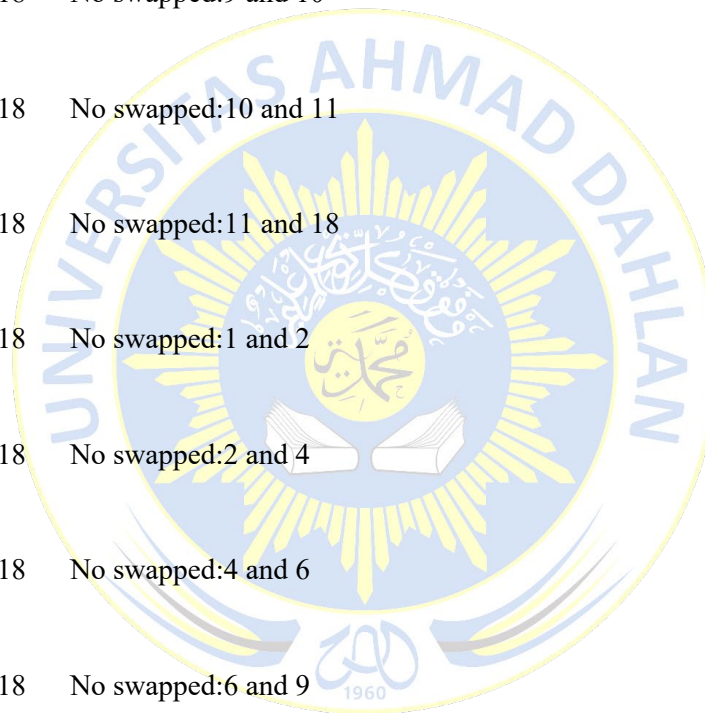
Pass: 48

1 2 4 6 9 10 11 18 No swapped:10 and 11

Pass: 49

1 2 4 6 9 10 11 18 No swapped:11 and 18

Pass: 50



1 2 4 6 9 10 11 18 No swapped:1 and 2

Pass: 51

1 2 4 6 9 10 11 18 No swapped:2 and 4

Pass: 52

1 2 4 6 9 10 11 18 No swapped:4 and 6

Pass: 53

1 2 4 6 9 10 11 18 No swapped:6 and 9

Pass: 54

1 2 4 6 9 10 11 18 No swapped:9 and 10

Pass: 55

1 2 4 6 9 10 11 18 No swapped:10 and 11

Pass: 56

1 2 4 6 9 10 11 18 No swapped:11 and 18

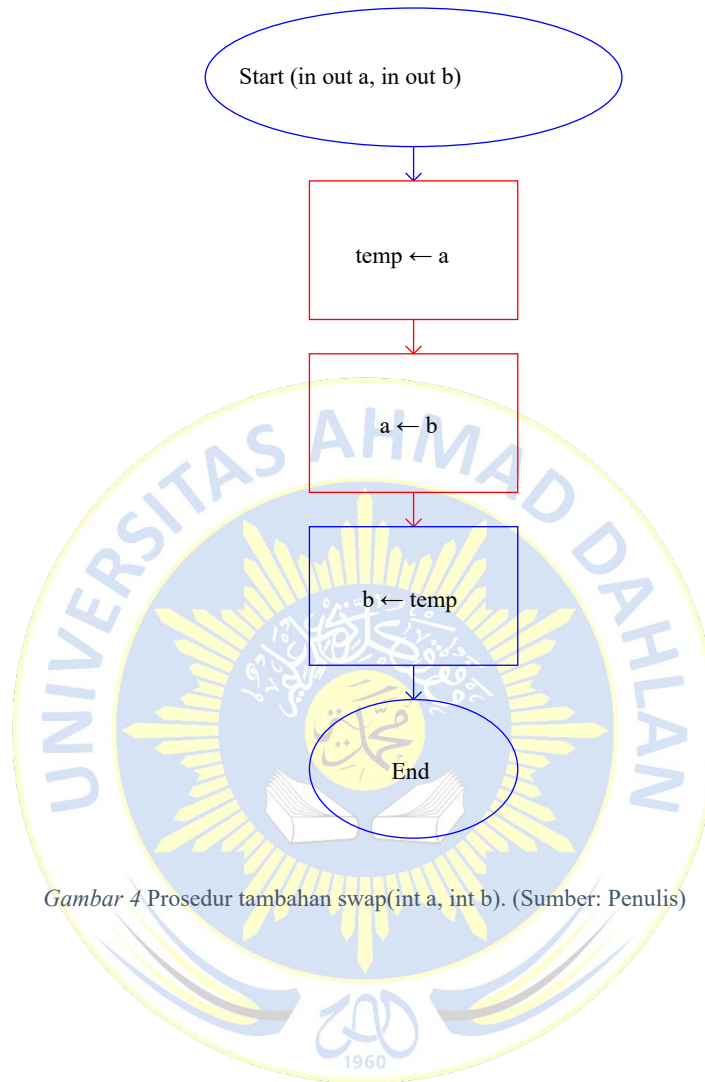
Unsorted Series:

1 2 4 6 9 10 11 18

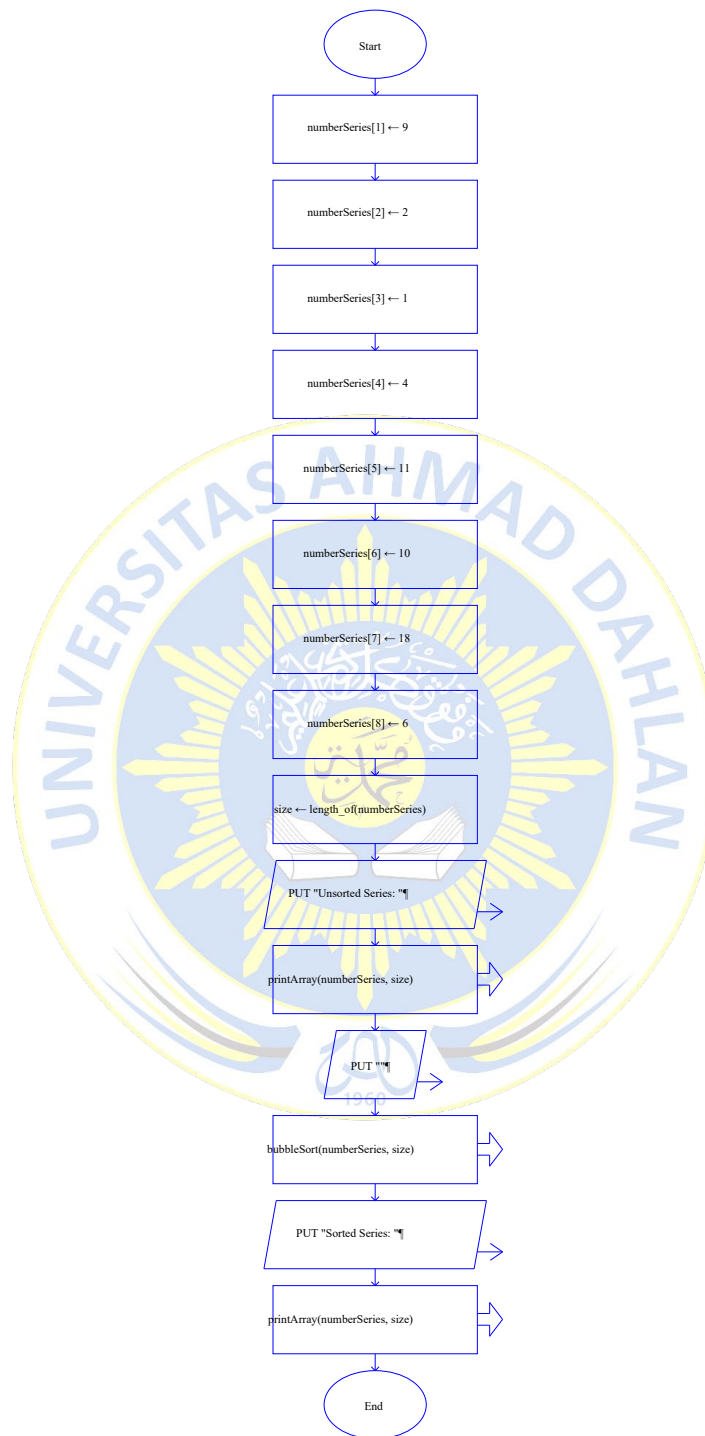
----Run complete. 2648 symbols evaluated.----

2. Seperti nomor 1, gunakan subprogram dalam flowchart untuk mengurutkan deret bilangan **9 2 1 4 11 10 18 6** dengan bubble sort dan tampilkan tiap langkah sortingnya secara manual.

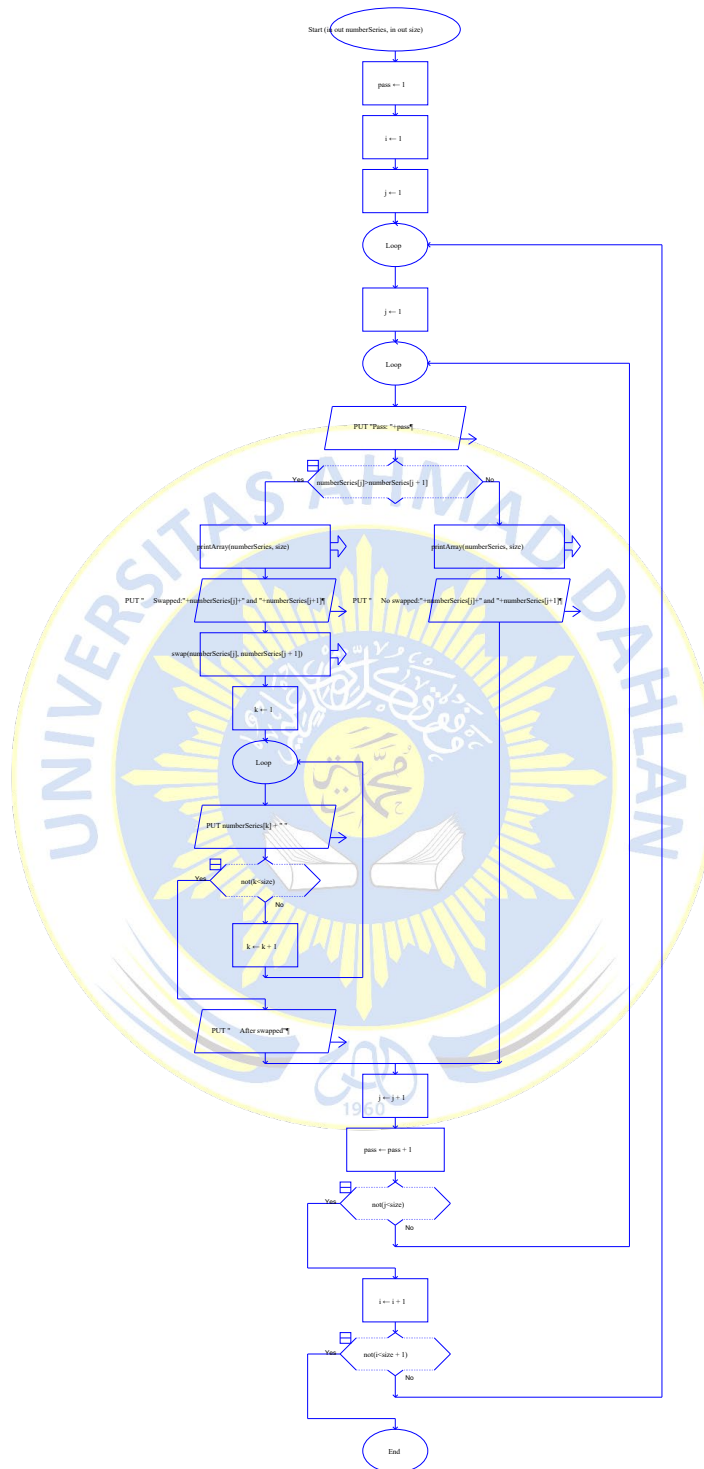
Berikut adalah flowchart mengurutkan dengan algoritma bubble sort.



Gambar 4 Prosedur tambahan swap(int a, int b). (Sumber: Penulis)



Gambar 5 Fungsi utama untuk mendeklarasikan variabel dan memanggil subprogram-subprogram. (Sumber: Penulis)



Gambar 6 Prosedur bubbleSort(). (Sumber:Penulis)

Algoritma yang digunakan dalam flowchart ini adalah Bubble Sort, sama seperti sebelumnya. Namun, pada flowchart ini, kita menggunakan subProgram untuk memisahkan beberapa bagian kode yang berulang atau memiliki fungsionalitas yang spesifik.

Flowchart ini terdiri dari sebuah prosedur utama bernama bubbleSort-subProgram dan tiga prosedur bantuan, yaitu swap, printArray, dan bubbleSort. Prosedur swap dan bubbleSort memiliki fungsi yang sama seperti pada flowchart sebelumnya. Prosedur printArray digunakan untuk mencetak elemen-elemen dalam array yang diberikan sebagai input.

Alur prosedur utama bubbleSort-subProgram:

- 1) Mengisi array numberSeries dengan angka-angka yang akan diurutkan.
- 2) Menampilkan angka-angka dalam array numberSeries sebelum diurutkan menggunakan prosedur printArray.
- 3) Mengurutkan array numberSeries menggunakan prosedur bubbleSort.
- 4) Menampilkan angka-angka dalam array numberSeries setelah diurutkan menggunakan prosedur printArray.

Alur prosedur bubbleSort:

- 1) Menginisialisasi variabel pass, i, dan j dengan nilai 1.
- 2) Melakukan loop luar dengan variabel i sebagai penghitung. Loop ini akan berjalan sebanyak size - 1 kali.
- 3) Melakukan loop dalam dengan variabel j sebagai penghitung. Loop ini akan berjalan sebanyak size - 1 kali.
- 4) Menampilkan nilai pass saat ini.
- 5) Membandingkan elemen numberSeries(j) dengan elemen numberSeries(j+1). Jika elemen numberSeries(j) lebih besar dari elemen numberSeries(j+1), maka:
 - a. Menampilkan elemen yang akan ditukar menggunakan prosedur printArray.
 - b. Menukar elemen numberSeries(j) dengan elemen numberSeries(j+1) menggunakan prosedur swap.
 - c. Menampilkan elemen setelah ditukar menggunakan prosedur printArray.
- 6) Jika elemen numberSeries(j) tidak lebih besar dari elemen numberSeries(j+1), maka menampilkan bahwa tidak ada penukaran elemen menggunakan prosedur printArray.
- 7) Mengincrement nilai j dan pass.
- 8) Mengulangi langkah 4-7 hingga loop dalam selesai.
- 9) Mengincrement nilai i.
- 10) Mengulangi langkah 3-9 hingga loop luar selesai.

Dengan menggunakan subProgram, kode menjadi lebih modular dan mudah untuk dibaca dan dikelola. Algoritma Bubble Sort yang digunakan tetap sama, dan array numberSeries akan diurutkan secara ascending.

Berikut adalah outputnya:



Gambar 7 Output dengan membuat subProgram bubbleSort(). (Sumber: Penulis)

Berikut adalah output berbentuk logs teks pada raptor:

Unsorted Series:

9 2 1 4 11 10 18 6

Pass: 1

9 2 1 4 11 10 18 6 Swapped:9 and 2

2 9 1 4 11 10 18 6 After swapped

Pass: 2

2 9 1 4 11 10 18 6 Swapped:9 and 1

2 1 9 4 11 10 18 6 After swapped

Pass: 3

2 1 9 4 11 10 18 6 Swapped:9 and 4

2 1 4 9 11 10 18 6 After swapped

Pass: 4

2 1 4 9 11 10 18 6 No swapped:9 and 11

Pass: 5

2 1 4 9 11 10 18 6 Swapped:11 and 10

2 1 4 9 10 11 18 6 After swapped

Pass: 6

2 1 4 9 10 11 18 6 No swapped:11 and 18

Pass: 7

2 1 4 9 10 11 18 6 Swapped:18 and 6

2 1 4 9 10 11 6 18 After swapped

Pass: 8

2 1 4 9 10 11 6 18 Swapped:2 and 1

1 2 4 9 10 11 6 18 After swapped

Pass: 9

1 2 4 9 10 11 6 18 No swapped:2 and 4

Pass: 10

1 2 4 9 10 11 6 18 No swapped:4 and 9

Pass: 11



1 2 4 9 10 11 6 18 No swapped:9 and 10

Pass: 12

1 2 4 9 10 11 6 18 No swapped:10 and 11

Pass: 13

1 2 4 9 10 11 6 18 Swapped:11 and 6

1 2 4 9 10 6 11 18 After swapped

Pass: 14

1 2 4 9 10 6 11 18 No swapped:11 and 18

Pass: 15

1 2 4 9 10 6 11 18 No swapped:1 and 2

Pass: 16

1 2 4 9 10 6 11 18 No swapped:2 and 4

Pass: 17

1 2 4 9 10 6 11 18 No swapped:4 and 9

Pass: 18

1 2 4 9 10 6 11 18 No swapped:9 and 10

Pass: 19

1 2 4 9 10 6 11 18 Swapped:10 and 6

1 2 4 9 6 10 11 18 After swapped

Pass: 20

1 2 4 9 6 10 11 18 No swapped:10 and 11

Pass: 21

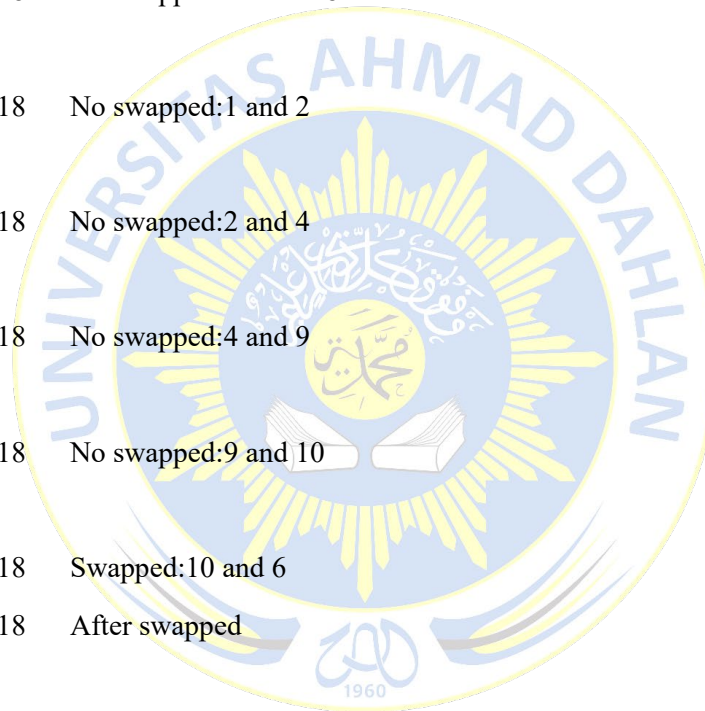
1 2 4 9 6 10 11 18 No swapped:11 and 18

Pass: 22

1 2 4 9 6 10 11 18 No swapped:1 and 2

Pass: 23

1 2 4 9 6 10 11 18 No swapped:2 and 4



Pass: 24

1 2 4 9 6 10 11 18 No swapped:4 and 9

Pass: 25

1 2 4 9 6 10 11 18 Swapped:9 and 6

1 2 4 6 9 10 11 18 After swapped

Pass: 26

1 2 4 6 9 10 11 18 No swapped:9 and 10

Pass: 27

1 2 4 6 9 10 11 18 No swapped:10 and 11

Pass: 28

1 2 4 6 9 10 11 18 No swapped:11 and 18

Pass: 29

1 2 4 6 9 10 11 18 No swapped:1 and 2

Pass: 30

1 2 4 6 9 10 11 18 No swapped:2 and 4

Pass: 31

1 2 4 6 9 10 11 18 No swapped:4 and 6

Pass: 32

1 2 4 6 9 10 11 18 No swapped:6 and 9

Pass: 33

1 2 4 6 9 10 11 18 No swapped:9 and 10

Pass: 34

1 2 4 6 9 10 11 18 No swapped:10 and 11

Pass: 35

1 2 4 6 9 10 11 18 No swapped:11 and 18

Pass: 36

1 2 4 6 9 10 11 18 No swapped:1 and 2



Pass: 37

1 2 4 6 9 10 11 18 No swapped:2 and 4

Pass: 38

1 2 4 6 9 10 11 18 No swapped:4 and 6

Pass: 39

1 2 4 6 9 10 11 18 No swapped:6 and 9

Pass: 40

1 2 4 6 9 10 11 18 No swapped:9 and 10

Pass: 41

1 2 4 6 9 10 11 18 No swapped:10 and 11

Pass: 42

1 2 4 6 9 10 11 18 No swapped:11 and 18

Pass: 43

1 2 4 6 9 10 11 18 No swapped:1 and 2

Pass: 44

1 2 4 6 9 10 11 18 No swapped:2 and 4

Pass: 45

1 2 4 6 9 10 11 18 No swapped:4 and 6

Pass: 46

1 2 4 6 9 10 11 18 No swapped:6 and 9

Pass: 47

1 2 4 6 9 10 11 18 No swapped:9 and 10

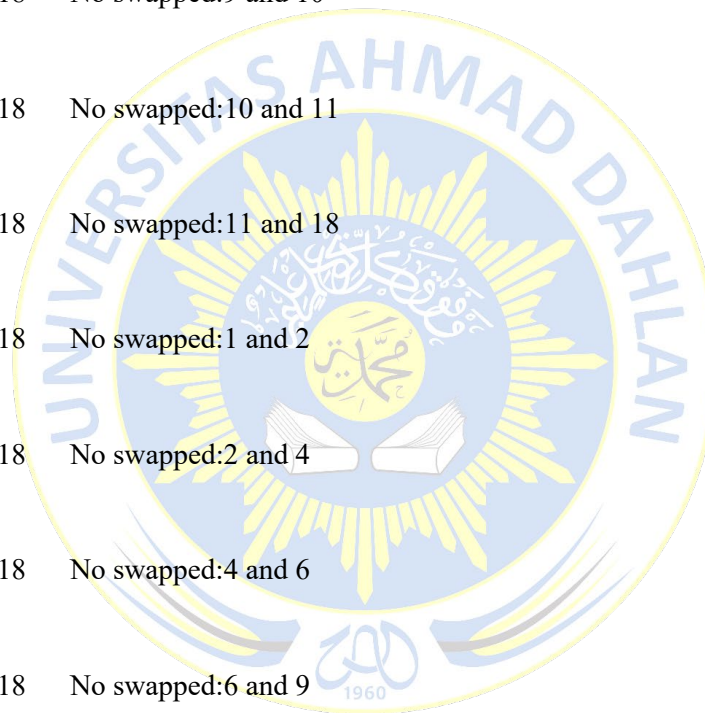
Pass: 48

1 2 4 6 9 10 11 18 No swapped:10 and 11

Pass: 49

1 2 4 6 9 10 11 18 No swapped:11 and 18

Pass: 50



1 2 4 6 9 10 11 18 No swapped:1 and 2

Pass: 51

1 2 4 6 9 10 11 18 No swapped:2 and 4

Pass: 52

1 2 4 6 9 10 11 18 No swapped:4 and 6

Pass: 53

1 2 4 6 9 10 11 18 No swapped:6 and 9

Pass: 54

1 2 4 6 9 10 11 18 No swapped:9 and 10

Pass: 55

1 2 4 6 9 10 11 18 No swapped:10 and 11

Pass: 56

1 2 4 6 9 10 11 18 No swapped:11 and 18

Sorted Series:

1 2 4 6 9 10 11 18

----Run complete. 2825 symbols evaluated.----

3. Konversikan hasil dari flowchart nomor 1 dan 2 menjadi program C++.

Konversi tanpa subProgram

Berikut adalah kodingan yang telah dibuat ke dalam C++ tetapi tanpa subprogram:

```

1  #include <iostream>
2  using namespace std;
3
4  int main(){
5      int numberSeries[]={9,2,1,4,11,10,18,6};
6      int size=sizeof(numberSeries)/sizeof(numberSeries[0]);
7      cout<<"Unsorted Series: "<<endl;
8      for(int i=0; i<size; i++) cout<<numberSeries[i]<<" ";
9      cout<<endl;
10
11     int pass=1;
12     for(int i=0; i<size; i++){
13         for(int j=0; j<size-1; j++){
14             cout<<"\nPass: "<<pass<<endl;
15             if(numberSeries[j]>numberSeries[j+1]){
16                 for(int k=0; k<size; k++){
17                     if(k==j || k==j+1) cout<<"\033[1;31m"<<numberSeries[k]<<"\033[0m"<<" ";
18                     else cout<<numberSeries[k]<<" ";
19                 }
20                 cout<<"    Swapped: "<<numberSeries[j]<<" and "<<numberSeries[j+1]<<endl;
21                 swap(numberSeries[j], numberSeries[j+1]);
22                 for(int k=0; k<size; k++){
23                     if(k==j || k==j+1) cout<<"\033[1;32m"<<numberSeries[k]<<"\033[0m"<<" ";
24                     else cout<<numberSeries[k]<<" ";
25                 }
26                 cout<<"    After swapped"<<endl;
27             }
28             else{
29                 for(int k=0; k<size; k++){
30                     if(k==j || k==j+1) cout<<"\033[1;32m"<<numberSeries[k]<<"\033[0m"<<" ";
31                     else cout<<numberSeries[k]<<" ";
32                 }
33                 cout<<"    No swapped: "<<numberSeries[j]<<" and "<<numberSeries[j+1]<<endl;
34             }
35             pass++;
36         }
37     }
38     cout<<"\nSorted Series: "<<endl;
39     for(int i=0; i<size; i++){
40         cout << numberSeries[i] << " ";
41     }
42     cout<<endl;
43     return 0;
44 }

```

Gambar 8 Kodingan mengurutkan dengan bubbleSort tanpa subProgram. (Sumber: Penulis)

Program ini menggunakan algoritma Bubble Sort untuk mengurutkan array numberSeries. Program ini tidak menggunakan subProgram, sehingga semua kode ada dalam fungsi main.

Alur program:

- 1) Mendefinisikan array numberSeries yang berisi angka-angka yang akan diurutkan.
- 2) Menghitung ukuran array numberSeries dan menyimpannya dalam variabel size.
- 3) Menampilkan angka-angka dalam array numberSeries sebelum diurutkan.
- 4) Menginisialisasi variabel pass dengan nilai 1.

- 5) Melakukan loop luar dengan variabel *i* sebagai penghitung. Loop ini akan berjalan sebanyak *size* kali.
- 6) Melakukan loop dalam dengan variabel *j* sebagai penghitung. Loop ini akan berjalan sebanyak *size* - 1 kali.
- 7) Menampilkan nilai *pass* saat ini.
- 8) Membandingkan elemen `numberSeries[j]` dengan elemen `numberSeries[j+1]`. Jika elemen `numberSeries[j]` lebih besar dari elemen `numberSeries[j+1]`, maka: a. Menampilkan elemen yang akan ditukar dengan warna merah (menggunakan ANSI escape code). b. Menampilkan pesan bahwa elemen akan ditukar. c. Menukar elemen `numberSeries[j]` dengan elemen `numberSeries[j+1]` menggunakan fungsi `swap` bawaan C++. d. Menampilkan elemen setelah ditukar dengan warna hijau (menggunakan ANSI escape code). e. Menampilkan pesan bahwa elemen telah ditukar.
- 9) Jika elemen `numberSeries[j]` tidak lebih besar dari elemen `numberSeries[j+1]`, maka: a. Menampilkan elemen yang tidak ditukar dengan warna hijau (menggunakan ANSI escape code). b. Menampilkan pesan bahwa tidak ada penukaran elemen.
- 10) Mengincrement nilai *j* dan *pass*.
- 11) Mengulangi langkah 7-10 hingga loop dalam selesai.
- 12) Mengincrement nilai *i*.
- 13) Mengulangi langkah 6-12 hingga loop luar selesai.
- 14) Menampilkan angka-angka dalam array `numberSeries` setelah diurutkan.
- 15) Mengakhiri program dengan `return 0`.

Fitur-fitur dalam program ini:

- Menggunakan algoritma Bubble Sort untuk mengurutkan array.
- Menampilkan elemen-elemen array sebelum dan sesudah diurutkan.
- Menampilkan setiap langkah penukaran elemen dalam proses pengurutan.
- Menggunakan ANSI escape code untuk menampilkan elemen yang ditukar atau tidak ditukar dengan warna yang berbeda (merah dan hijau).

Program ini mengurutkan array `numberSeries` secara ascending dengan algoritma Bubble Sort. Selama proses pengurutan, program menampilkan setiap langkah penukaran elemen dan elemen yang tidak ditukar dengan warna yang berbeda untuk memudahkan pemahaman.

Berikut adalah outputnya:

```

PS D:\Document\MIKKVuliah\Semester 2\Algoritma Pemrograman\Praktikum\Pertemuan 7> cd "d:\Document\MIKKVuliah\Semester 2\Algoritma Pemrograman\Praktikum\Pertemuan 7" ; if ($?) { g++ -O3 "bubbleSort_withoutSubProgram.cpp" -o "bubbleSort_withoutSubProgram.exe" } ; if ($?) { & ".\bubbleSort_withoutSubProgram.exe" }
Unsorted Sorting:
9 2 1 4 11 10 18 6

Pass: 1
9 2 1 4 11 10 18 6   Swapped: 9 and 2
2 9 1 4 11 10 18 6   After swapped

Pass: 2
2 9 1 4 11 10 18 6   Swapped: 9 and 1
2 1 9 4 11 10 18 6   After swapped

Pass: 3
2 1 9 4 11 10 18 6   Swapped: 9 and 4
2 1 4 9 11 10 18 6   After swapped

Pass: 4
2 1 4 9 11 10 18 6   No swapped: 9 and 11

Pass: 5
2 1 4 9 11 10 18 6   Swapped: 11 and 10
2 1 4 9 10 11 18 6   After swapped

Pass: 6
2 1 4 9 10 11 18 6   No swapped: 11 and 18

Pass: 7
2 1 4 9 10 11 18 6   Swapped: 10 and 6
2 1 4 9 10 11 6 18   After swapped

Pass: 8
2 1 4 9 10 11 6 18   Swapped: 2 and 1
1 2 4 9 10 11 6 18   After swapped

Pass: 9
1 2 4 9 10 11 6 18   No swapped: 2 and 4

Pass: 10
1 2 4 9 10 11 6 18   No swapped: 4 and 9

Pass: 11
1 2 4 9 10 11 6 18   No swapped: 9 and 10

Pass: 12
1 2 4 9 10 11 6 18   No swapped: 10 and 11

Pass: 13
1 2 4 9 10 11 6 18   Swapped: 11 and 6
1 2 4 9 10 6 11 18   After swapped

Pass: 14
1 2 4 9 10 6 11 18   No swapped: 11 and 18

Pass: 15
1 2 4 9 10 6 11 18   No swapped: 1 and 2

Pass: 16
1 2 4 9 10 6 11 18   No swapped: 1 and 2
  
```

```

Pass: 15
1 2 4 9 10 6 11 18   No swapped: 1 and 2

Pass: 16
1 2 4 9 10 6 11 18   No swapped: 2 and 4

Pass: 17
1 2 4 9 10 6 11 18   No swapped: 4 and 9

Pass: 18
1 2 4 9 10 6 11 18   No swapped: 9 and 10

Pass: 19
1 2 4 9 10 6 11 18   Swapped: 10 and 6
1 2 4 9 6 10 11 18   After swapped

Pass: 20
1 2 4 9 6 10 11 18   No swapped: 10 and 11

Pass: 21
1 2 4 9 6 10 11 18   No swapped: 11 and 18

Pass: 22
1 2 4 9 6 10 11 18   No swapped: 1 and 2

Pass: 23
1 2 4 9 6 10 11 18   No swapped: 2 and 4

Pass: 24
1 2 4 9 6 10 11 18   No swapped: 4 and 9

Pass: 25
1 2 4 6 9 10 11 18   Swapped: 9 and 6
1 2 4 6 9 10 11 18   After swapped

Pass: 26
1 2 4 6 9 10 11 18   No swapped: 9 and 10

Pass: 27
1 2 4 6 9 10 11 18   No swapped: 10 and 11

Pass: 28
1 2 4 6 9 10 11 18   No swapped: 11 and 18

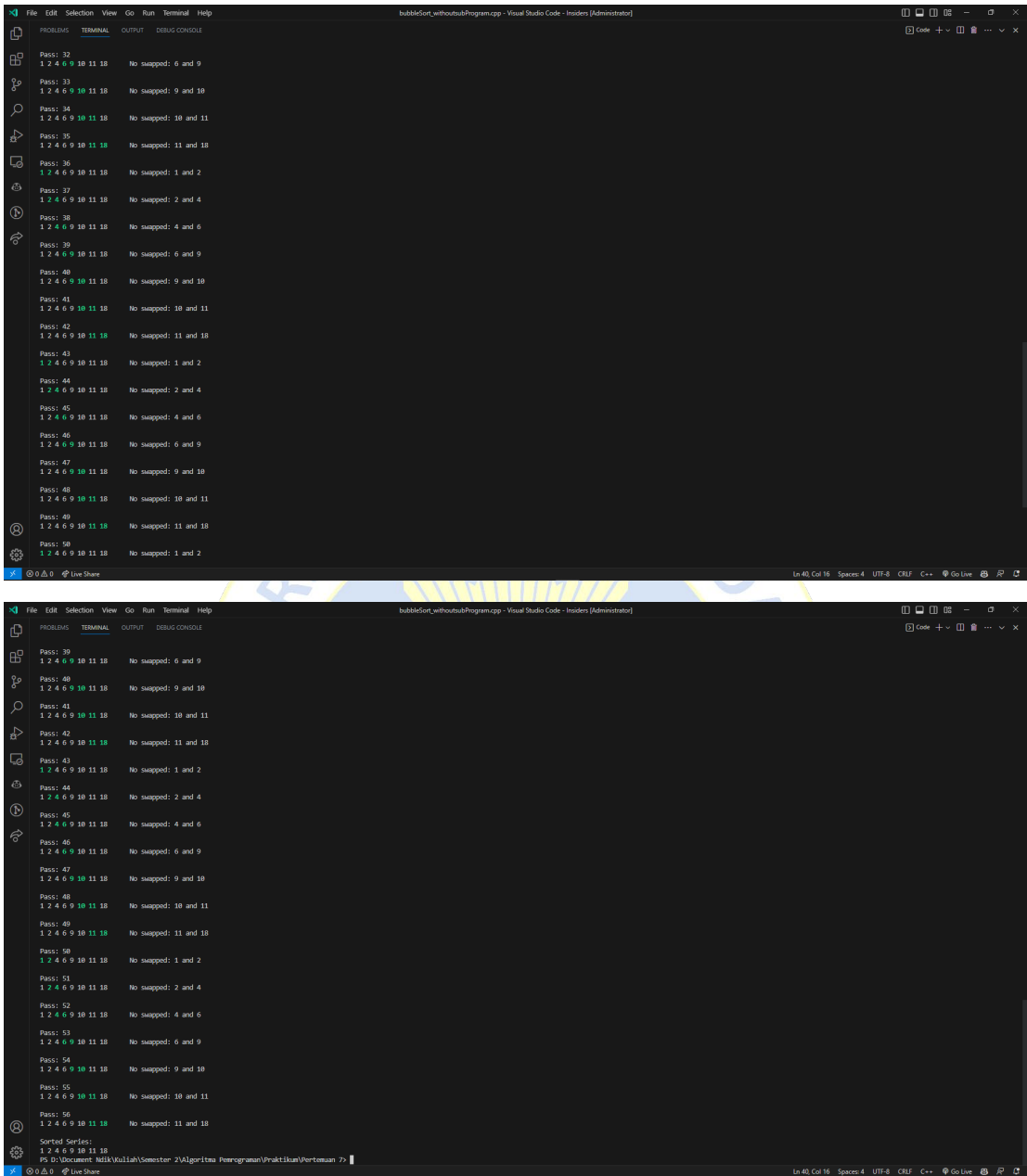
Pass: 29
1 2 4 6 9 10 11 18   No swapped: 1 and 2

Pass: 30
1 2 4 6 9 10 11 18   No swapped: 2 and 4

Pass: 31
1 2 4 6 9 10 11 18   No swapped: 4 and 6

Pass: 32
1 2 4 6 9 10 11 18   No swapped: 6 and 9

Pass: 33
1 2 4 6 9 10 11 18   No swapped: 6 and 9
  
```



Gambar 9 Output program. (Sumber: Penulis)

Konversi dengan subProgram

Berikut adalah program mengurutkan array dengan algoritma bubbleSort dan menggunakan subprogram.

```

1  #include <iostream>
2  using namespace std;
3
4  void printArray(int numberSeries[], int size){
5      for(int i=0; i<size; i++) cout<<numberSeries[i]<<" ";
6  }
7
8  void bubbleSort(int numberSeries[], int size){
9      int pass=1;
10     for(int i=0; i<size; i++){
11         for(int j=0; j<size-1; j++){
12             cout<<"\nPass: "<<pass<<endl;
13             if(numberSeries[j]>numberSeries[j+1]){
14                 for(int k=0; k<size; k++){
15                     if(k==j || k==j+1) cout<<"\033[1;31m"<<numberSeries[k]<<"\033[0m"<<" ";
16                     else cout<<numberSeries[k]<<" ";
17                     cout<<"    Swapped: "<<numberSeries[j]<<" and "<<numberSeries[j+1]<<endl;
18                     swap(numberSeries[j], numberSeries[j+1]);
19                     for(int k=0; k<size; k++){
20                         if(k==j || k==j+1) cout<<"\033[1;32m"<<numberSeries[k]<<"\033[0m"<<" ";
21                         else cout<<numberSeries[k]<<" ";
22                         cout<<"    After swapped"<<endl;
23                     }
24                 }
25             }
26             else{
27                 for(int k=0; k<size; k++){
28                     if(k==j || k==j+1) cout<<"\033[1;32m"<<numberSeries[k]<<"\033[0m"<<" ";
29                     else cout<<numberSeries[k]<<" ";
30                     cout<<"    No swapped: "<<numberSeries[j]<<" and "<<numberSeries[j+1]<<endl;
31                 }
32             }
33             pass++;
34         }
35     }
36 }
37
38 int main(){
39     int numberSeries[]={9,2,1,4,11,10,18,6};
40     int size=sizeof(numberSeries)/sizeof(numberSeries[0]);
41     cout<<"Unsorted Series: "<<endl;
42     printArray(numberSeries, size);
43     bubbleSort(numberSeries, size);
44     cout<<"\nSorted Series: "<<endl;
45     for(int i=0; i<size; i++){
46         cout << numberSeries[i] << " ";
47     }
48 }

```

Gambar 10 Menggunakan subProgram dalam program C++. (Sumber: Penulis)

Berikut adalah penjelasan alur program C++ yang menggunakan subProgram:

Program ini menggunakan algoritma Bubble Sort untuk mengurutkan array numberSeries. Program ini menggunakan subProgram, yaitu printArray dan bubbleSort, untuk memisahkan beberapa bagian kode yang berulang atau memiliki fungsionalitas yang spesifik.

Alur program:

- 1) Mendefinisikan fungsi printArray yang menerima array numberSeries dan ukuran array size sebagai parameter. Fungsi ini mencetak elemen-elemen dalam array yang diberikan.
- 2) Mendefinisikan fungsi bubbleSort yang menerima array numberSeries dan ukuran array size sebagai parameter. Fungsi ini mengurutkan array menggunakan algoritma Bubble Sort dan menampilkan setiap langkah penukaran elemen dalam proses pengurutan.
- 3) Mendefinisikan array numberSeries yang berisi angka-angka yang akan diurutkan dalam fungsi main.
- 4) Menghitung ukuran array numberSeries dan menyimpannya dalam variabel size.
- 5) Menampilkan angka-angka dalam array numberSeries sebelum diurutkan menggunakan fungsi printArray.
- 6) Mengurutkan array numberSeries menggunakan fungsi bubbleSort.
- 7) Menampilkan angka-angka dalam array numberSeries setelah diurutkan menggunakan fungsi printArray.

Fitur-fitur dalam program ini:

- Menggunakan algoritma Bubble Sort untuk mengurutkan array.
- Menampilkan elemen-elemen array sebelum dan sesudah diurutkan.
- Menampilkan setiap langkah penukaran elemen dalam proses pengurutan.
- Menggunakan ANSI escape code untuk menampilkan elemen yang ditukar atau tidak ditukar dengan warna yang berbeda (merah dan hijau).
- Menggunakan subProgram untuk memisahkan beberapa bagian kode yang berulang atau memiliki fungsionalitas yang spesifik, sehingga kode menjadi lebih modular dan mudah untuk dibaca dan dikelola.

Program ini mengurutkan array numberSeries secara ascending dengan algoritma Bubble Sort. Selama proses pengurutan, program menampilkan setiap langkah penukaran elemen dan elemen yang tidak ditukar dengan warna yang berbeda untuk memudahkan pemahaman. Dengan menggunakan subProgram, kode menjadi lebih modular dan mudah untuk dibaca dan dikelola.

Berikut adalah output programnya:

```

PS D:\Document NIKK\Kuliah\Semester 2\Algoritma Pemrograman\Praktikum\Portemon 7> cd "d:\Document NIKK\Kuliah\Semester 2\Algoritma Pemrograman\Praktikum\Portemon 7" ; if ($?) { g++ -O3 "bubbleSort_subProgram.cpp" -o "bubbleSort_subProgram.exe" } ; if ($?) {
& ".\bubbleSort_subProgram.exe" }
Unsorted Serial:
9 2 1 4 11 10 18 6
Pass: 1
9 2 1 4 11 10 18 6      Swapped: 9 and 2
2 9 1 4 11 10 18 6      After swapped

Pass: 2
2 9 1 4 11 10 18 6      Swapped: 9 and 1
2 1 9 4 11 10 18 6      After swapped

Pass: 3
2 1 9 4 11 10 18 6      Swapped: 9 and 4
2 1 4 9 11 10 18 6      After swapped

Pass: 4
2 1 4 9 11 10 18 6      No swapped: 9 and 11

Pass: 5
2 1 4 9 11 10 18 6      Swapped: 11 and 18
2 1 4 9 10 11 18 6      After swapped

Pass: 6
2 1 4 9 10 11 18 6      No swapped: 11 and 18

Pass: 7
2 1 4 9 10 11 18 6      Swapped: 18 and 6
2 1 4 9 10 11 6 18      After swapped

Pass: 8
2 1 4 9 10 11 6 18      Swapped: 2 and 1
1 2 4 9 10 11 6 18      After swapped

Pass: 9
1 2 4 9 10 11 6 18      No swapped: 2 and 4

Pass: 10
1 2 4 9 10 11 6 18      No swapped: 4 and 9

Pass: 11
1 2 4 9 10 11 6 18      No swapped: 9 and 10

Pass: 12
1 2 4 9 10 11 6 18      No swapped: 10 and 11

Pass: 13
1 2 4 9 10 11 6 18      Swapped: 11 and 6
1 2 4 9 10 6 11 18      After swapped

Pass: 14
1 2 4 9 10 6 11 18      No swapped: 11 and 18

Pass: 15
1 2 4 9 10 6 11 18      No swapped: 1 and 2

Pass: 16
1 2 4 9 10 6 11 18      No swapped: 2 and 4

Pass: 17
1 2 4 9 10 6 11 18      No swapped: 4 and 9

Pass: 18
1 2 4 9 10 6 11 18      No swapped: 9 and 10

Pass: 19
1 2 4 9 10 6 11 18      Swapped: 10 and 6
1 2 4 9 6 10 11 18      After swapped

Pass: 20
1 2 4 9 6 10 11 18      No swapped: 10 and 11

Pass: 21
1 2 4 9 6 10 11 18      No swapped: 11 and 18

Pass: 22
1 2 4 9 6 10 11 18      No swapped: 1 and 2

Pass: 23
1 2 4 9 6 10 11 18      No swapped: 2 and 4

Pass: 24
1 2 4 9 6 10 11 18      No swapped: 4 and 9

Pass: 25
1 2 4 9 6 10 11 18      Swapped: 9 and 6
1 2 4 6 9 10 11 18      After swapped

Pass: 26
1 2 4 6 9 10 11 18      No swapped: 9 and 10

Pass: 27
1 2 4 6 9 10 11 18      No swapped: 10 and 11

Pass: 28
1 2 4 6 9 10 11 18      No swapped: 11 and 18

Pass: 29
1 2 4 6 9 10 11 18      No swapped: 1 and 2

Pass: 30
1 2 4 6 9 10 11 18      No swapped: 2 and 4

Pass: 31
1 2 4 6 9 10 11 18      No swapped: 4 and 6

Pass: 32
1 2 4 6 9 10 11 18      No swapped: 6 and 9

Pass: 33
1 2 4 6 9 10 11 18
  
```

The image displays two screenshots of a Visual Studio Code terminal window showing the output of a bubble sort program. The terminal output is as follows:

```

Pass: 32
1 2 4 6 9 10 11 18      No swapped: 6 and 9

Pass: 33
1 2 4 6 9 10 11 18      No swapped: 9 and 10

Pass: 34
1 2 4 6 9 10 11 18      No swapped: 10 and 11

Pass: 35
1 2 4 6 9 10 11 18      No swapped: 11 and 18

Pass: 36
1 2 4 6 9 10 11 18      No swapped: 1 and 2

Pass: 37
1 2 4 6 9 10 11 18      No swapped: 2 and 4

Pass: 38
1 2 4 6 9 10 11 18      No swapped: 4 and 6

Pass: 39
1 2 4 6 9 10 11 18      No swapped: 6 and 9

Pass: 40
1 2 4 6 9 10 11 18      No swapped: 9 and 10

Pass: 41
1 2 4 6 9 10 11 18      No swapped: 10 and 11

Pass: 42
1 2 4 6 9 10 11 18      No swapped: 11 and 18

Pass: 43
1 2 4 6 9 10 11 18      No swapped: 1 and 2

Pass: 44
1 2 4 6 9 10 11 18      No swapped: 2 and 4

Pass: 45
1 2 4 6 9 10 11 18      No swapped: 4 and 6

Pass: 46
1 2 4 6 9 10 11 18      No swapped: 6 and 9

Pass: 47
1 2 4 6 9 10 11 18      No swapped: 9 and 10

Pass: 48
1 2 4 6 9 10 11 18      No swapped: 10 and 11

Pass: 49
1 2 4 6 9 10 11 18      No swapped: 11 and 18

Pass: 50
1 2 4 6 9 10 11 18      No swapped: 1 and 2

Pass: 51
1 2 4 6 9 10 11 18      No swapped: 2 and 4

Pass: 52
1 2 4 6 9 10 11 18      No swapped: 4 and 6

Pass: 53
1 2 4 6 9 10 11 18      No swapped: 6 and 9

Pass: 54
1 2 4 6 9 10 11 18      No swapped: 9 and 10

Pass: 55
1 2 4 6 9 10 11 18      No swapped: 10 and 11

Pass: 56
1 2 4 6 9 10 11 18      No swapped: 11 and 18

Sorted Series:
1 2 4 6 9 10 11 18
PS D:\Document\idk\KuliaH\Semester 2\Algoritma Pemrograman\Praktikum\Portemon 7>
  
```

Gambar 11 Output program dengan subProgram. (Sumber: Penulis)

Untuk mengakses source kodingan maupun flowchart, dapat melihat link berikut:

<https://github.com/IRedDragonICY/Programming-Algorithms>