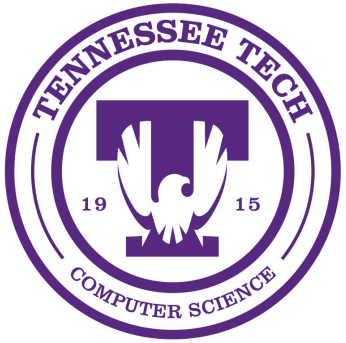
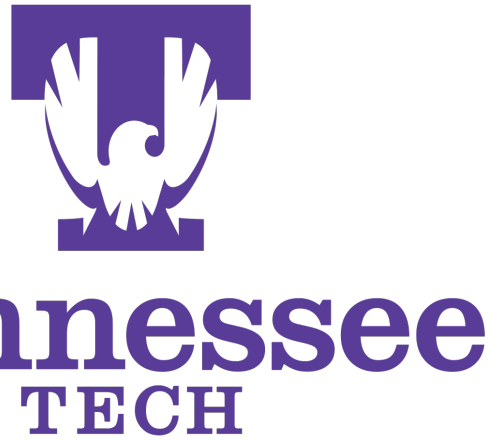
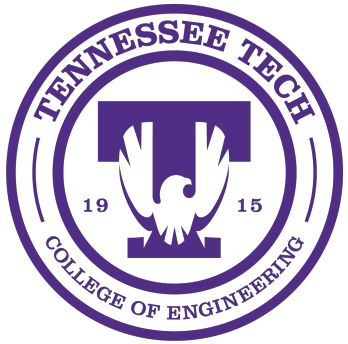






Branches and Flow Control

Ahsan Ayub

Ph.D. Student, Department of Computer Science

Graduate Research Assistant, CEROC

CSC 1300: Introduction to Programming

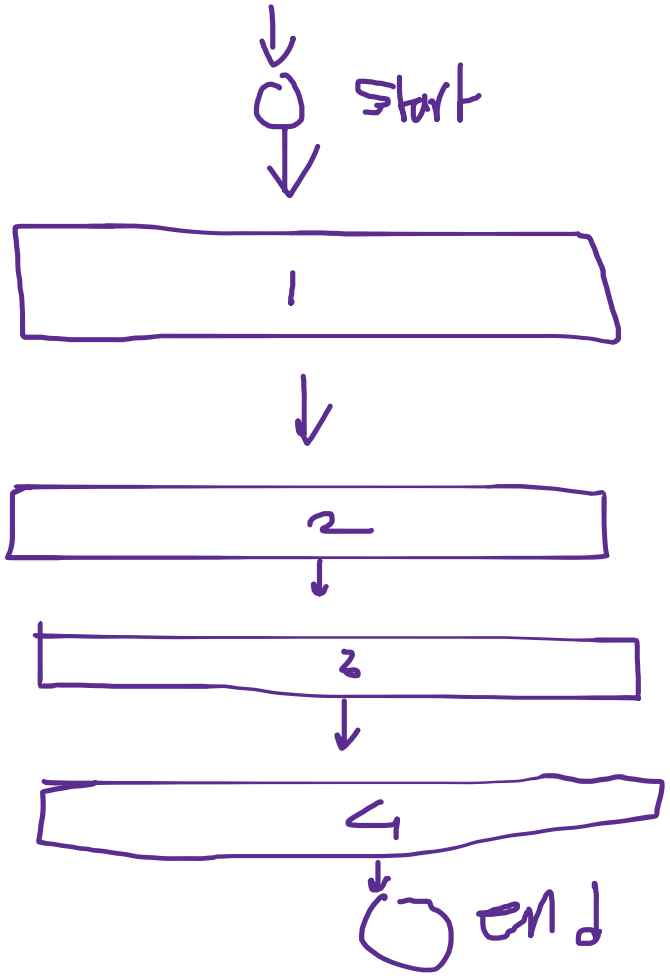
Wednesday, September 15, 2021

Flow Control

```
#include <iostream>
using namespace std;

int main()
{
        // Declaration and initialization
        int x = 5, y = 10, sum;
        sum = x + y;     // Addition
        cout << sum << endl;
        return 0;
}
```

Flow Control




```
#include <iostream>
using namespace std;

int main()
{
        // Declaration and initialization
      1 int x = 5, y = 10, sum;
      2 sum = x + y;     // Addition
      3 cout << sum << endl;
      4 return 0;
}
```

Flow Control

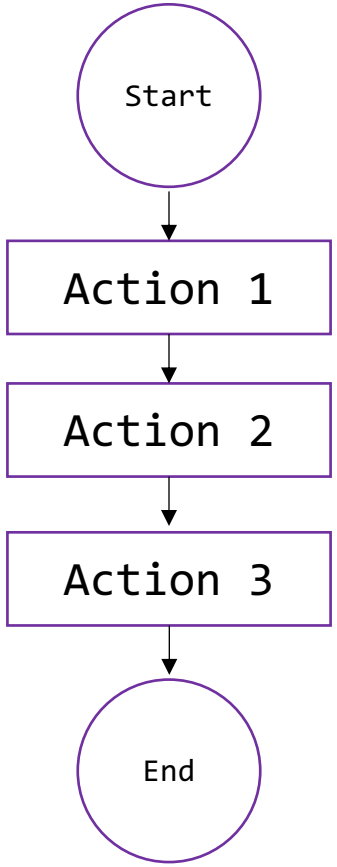
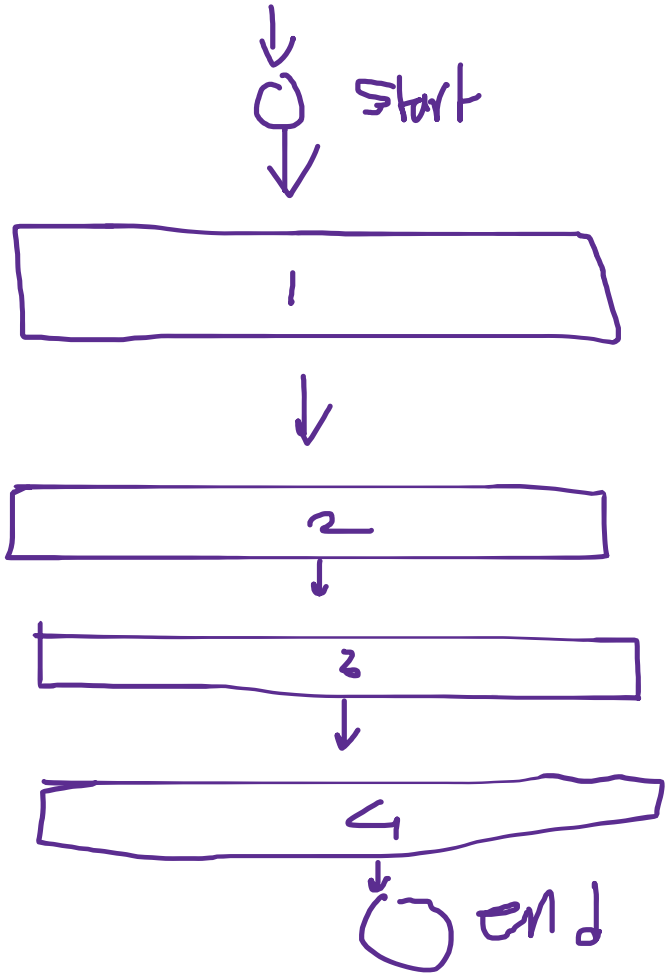


Fig. 1: Sequence-based flow control structure

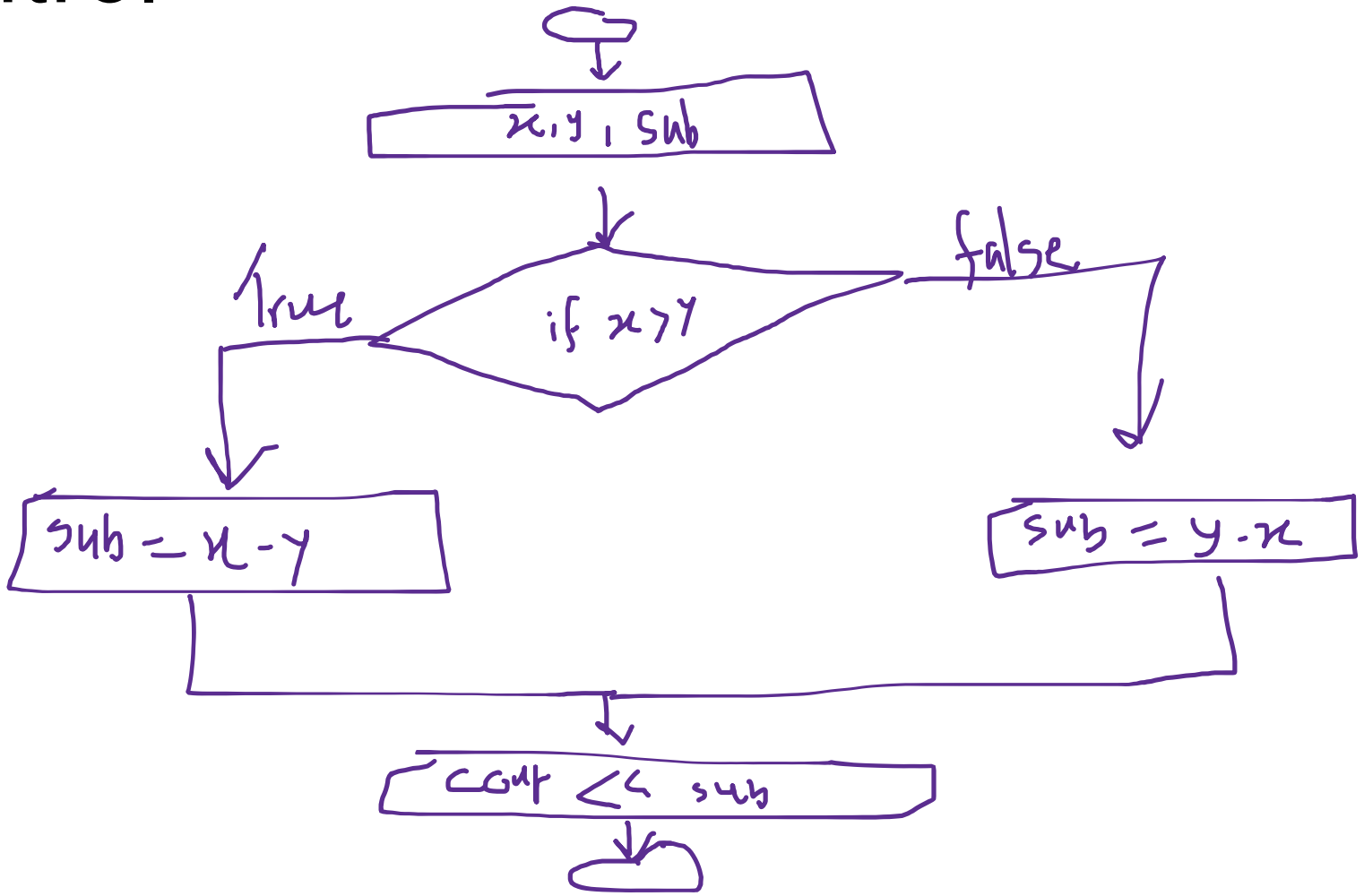
Flow Control




```
#include <iostream>
using namespace std;

int main()
{
        // Declaration and initialization
      1 int x = 5, y = 10, sub;
      2 sub = x - y;     // subtraction
      3 cout << sub << endl;
      4 return 0;
}
```

Flow Control



Flow Control

```
#include <iostream>
using namespace std;

int main()
{
        // Declaration and initialization
        int x = 5, y = 10;
        unsigned int sub;
        if(x > y)
        {
                sub = x - y;
        }
        else
        {
                 sub = x - y;
        }
        cout << sub << endl;
        return 0;
}
```

Flow Control

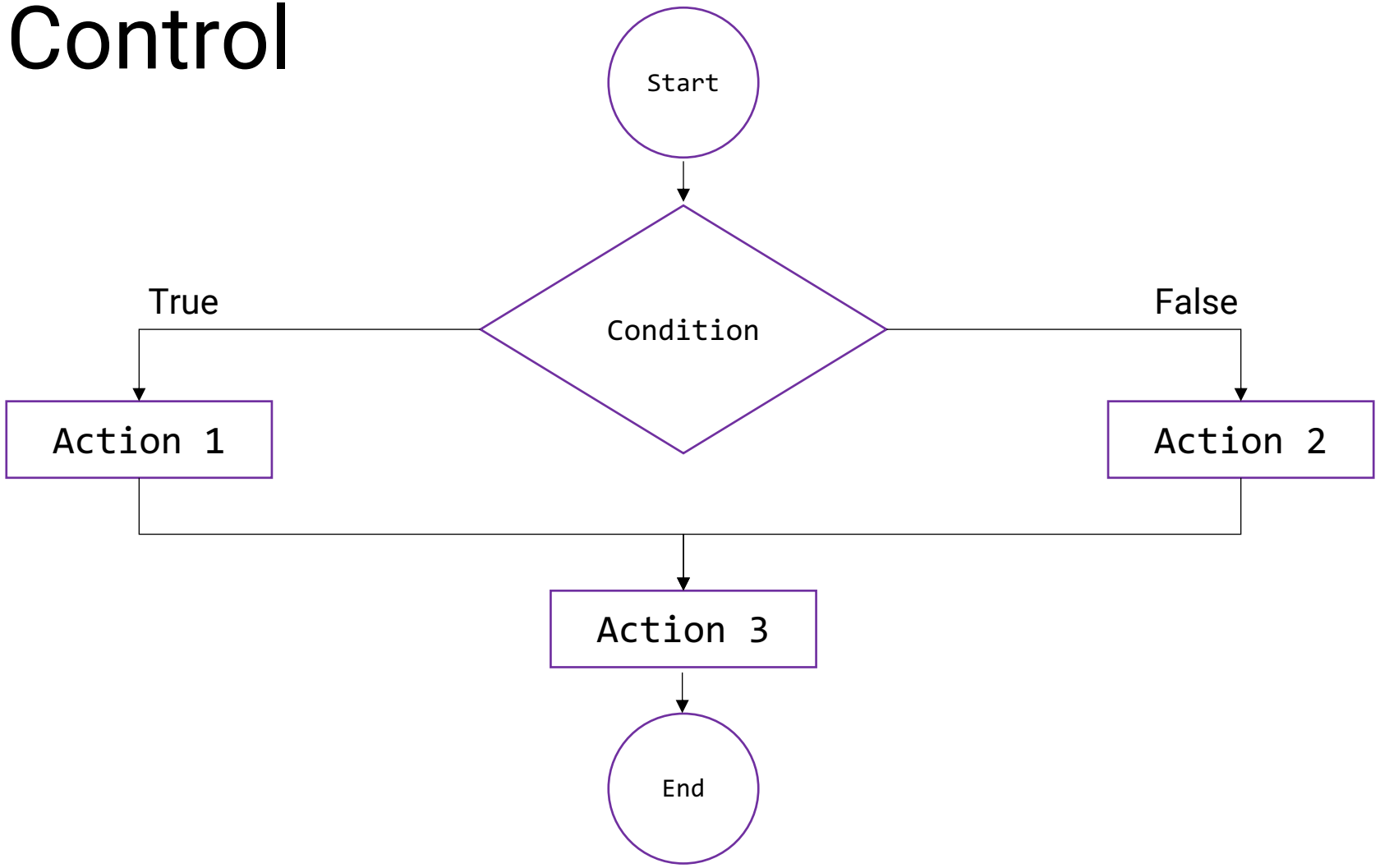


Fig. 2: Branch-based flow control structure

The `If` Statement

```
if (expression is true)
{
        action1;
}
action 2;
action 3;
```

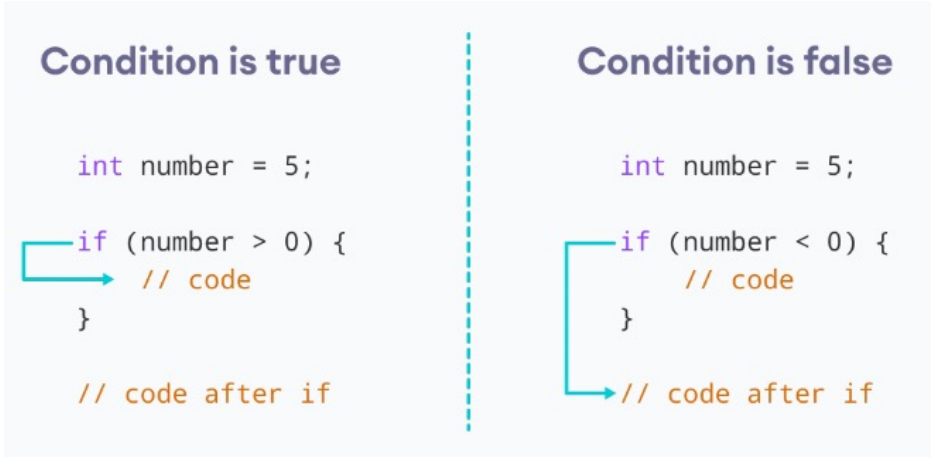
```
if (expression is true)
{
        action1;
}
else
{
        action 2;
}
action 3;
```

The `If` Statement

```
if (expression is true) {
        action1;
}
action 2;
action 3;
```

```
if (expression is true) {
        action1;
}
else {
        action 2;
}
action 3;
```

The `If` Statement



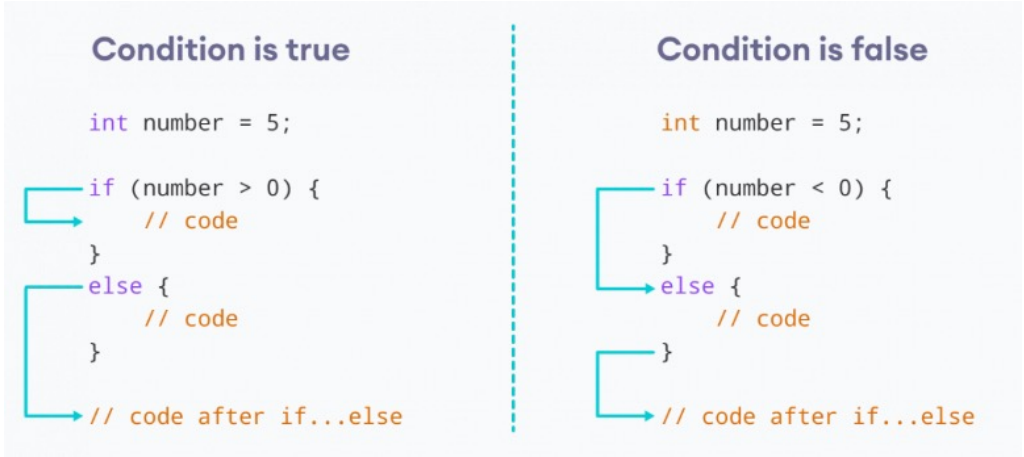


Image Source Programiz – https://www.programiz.com/cpp-programming/if-else

The `If` Statement – Output Tracing

```
#include <iostream>
using namespace std;

int main()
{
        // Declaration and initialization of the variables
        int a = 5, b = 10;
        if (a > b)
        {
                cout << "a is bigger than b" << endl;
        }
        if (b > a)
        {
                cout << ”b is bigger than a" << endl;
        }
        return 0;
}
```

```
??
```

The `If` Statement – Output Tracing

```
#include <iostream>
using namespace std;

int main()
{
        // Declaration and initialization of the variables
        int a = 5, b = 10;
        if (a > b)
        {
                cout << "a is bigger than b" << endl;
        }
        if (b > a)
        {
                cout << ”b is bigger than a" << endl;
        }
        return 0;
}
```

```
b is bigger than a
```

The `If` Statement – Output Tracing

```
#include <iostream>
using namespace std;

int main()
{
        // Declaration and initialization of the variables
        int a = 5, b = 10;
        if (a > b)
        {
                cout << "a is bigger than b" << endl;
        }
        else
        {
                cout << "b is bigger than a" << endl;
        }
        return 0;
}
```

```
b is bigger than a
```

Different Forms of `If` Statements

```
if (expression is true)
        action1;
```

```
if (expression is true)
{
        action1;
}
```

```
if (expression is true)
{
        action1;
        action2;
}
```

Must include the parathesis

```
if (expression is true)
        action1;
else
        action2;
```

```
if (expression is true)
{
        action1;
}
else
{
        action2;
}
```

```
if (expression is true)
{
        action1;
        action2;
}
else
{
        action3;
        action4;
}
```

Must include the parathesis

The `If` Statement – Output Tracing

```
#include <iostream>
using namespace std;

int main()
{
        // Declaration and initialization of the variables
        int a = 5, b = 10;
        if (a > b)
                cout << "a is bigger than b" << endl;
        if (b > a)
                cout << "b is bigger than a" << endl;
        if (a == b)
                cout << "a and b are equivalent" << endl;
        return 0;
}
```

```
??
```

The `else if` Statement

```
#include <iostream>
using namespace std;

int main()
{
        // Declaration and initialization of the variables
        int a = 20, b = 10;
        if (a == b)
                 cout << "a and b are equivalent" << endl;
        else if (a > b)
                cout << "a is bigger than b" << endl;
        else
                cout << "b is bigger than a" << endl;
        return 0;
}
```

```
??
```

The `else if` Statement

```
#include <iostream>
using namespace std;

int main()
{
        // Declaration and initialization of the variables
        int a = 20, b = 10;
        if (a == b)
                 cout << "a and b are equivalent" << endl;
        else if (a > b)
                cout << "a is bigger than b" << endl;
        else
                cout << "b is bigger than a" << endl;
        return 0;
}
```

```
a is bigger than b
```

The `else if` Statement – Output Tracing

```
#include <iostream>
using namespace std;

int main()
{
        // Declaration and initialization of the variables
        int a = 20, b = 10;
        if (a = b)
                 cout << "a and b are equivalent" << endl;
        else if (a > b)
                cout << "a is bigger than b" << endl;
        else
                cout << "b is bigger than a" << endl;
        return 0;
}
```

```
??
```

The `else if` Statement – Output Tracing

```
#include <iostream>
using namespace std;

int main()
{
        // Declaration and initialization of the variables
        int a = 20, b = 10;
        if (a = b)
                 cout << "a and b are equivalent" << endl;
        else if (a > b)
                cout << "a is bigger than b" << endl;
        else
                cout << "b is bigger than a" << endl;
        return 0;
}
```

```
a and b are equivalent
```

The `else if` Statement

1st Condition is true

```
int number = 2;
if (number > 0) {
    // code
}
else if (number == 0){
    // code
}
else {
    //code
}

//code after if
```

2nd Condition is true

```
int number = 0;
if (number > 0) {
    // code
}
else if (number == 0){
    // code
}
else {
    //code
}

//code after if
```

All Conditions are false

```
int number = -2;
if (number > 0) {
    // code
}
else if (number == 0){
    // code
}
else {
    //code
}

//code after if
```

Image Source Programiz – https://www.programiz.com/cpp-programming/if-else

Programming Challenge

Write a C++ program (on pen and paper) that will take an integer number from the user and display the number is positive, zero, or negative?

input

```
3
```

output

```
positive
```

input

```
0
```

output

```
zero
```

input

```
-54
```

output

```
negative
```

Programming Challenge – Solution

```
#include <iostream>
using namespace std;

int main()
{
        // Declaration the variable to store the user input
        int number;
        cin >> number;
        if (number < 0)  // Condition to check whether it's negative
                cout << "negative" << endl;
        else if (number == 0)    // Condition to check whether it's zero
                cout << "zero" << endl;
        else    // If it comes down here, then surely it's positive
                cout << "positive" << endl;
        return 0;
}
```

Programming Challenge

Write a C++ program (on pen and paper) that will take an integer number from the user and display the number is even (including zero) or odd?

input	input	input
3	62	0
output	output	output
odd	even	even

Programming Challenge – Solution

```
#include <iostream>
using namespace std;

int main()
{
        // Declaration the variable to store the user input
        int number;
        cin >> number;
        // Condition to check whether it's even
        if (number % 2 == 0)
                cout << "even" << endl;
         // If it comes down here, then surely it's odd
        else
                cout << "odd" << endl;
        return 0;
}
```

Programming Challenge

Write a C++ program (on pen and paper) that will take a character from the user and display whether the character is vowel and consonant?

input	output
a	vowel
D	consonant
I	vowel

Programming Challenge – Solution

```
#include <iostream>
using namespace std;

int main()
{
          // Declaration the variable to store the user input
          char character;
          cin >> character;
          if (character == ‘A’ || character == ‘a’)
                    cout << "vowel" << endl;
          else if (character == ‘E’ || character == ‘e’)
                    cout << "vowel" << endl;
          else if (character == ‘I’ || character == ‘i’)
                    cout << "vowel" << endl;
          else if (character == ‘O’ || character == ‘o’)
                    cout << "vowel" << endl;
          else if (character == ‘U’ || character == ‘u’)
                    cout << "vowel" << endl;
          else
                    cout << “consonant” << endl;
          return 0;
}
```

Programming Challenge – Follow-up

Write a C++ program (on pen and paper) that will take a character from the user and display whether the character is a vowel, a consonant, or unknown?

input	output
a	vowel
D	consonant
%	unknown

Programming Challenge – Solution

Introducing
Nested if..else

Programming Challenge – Solution

- Decimal representation of ‘A’ is 65 and ‘Z’ is 90
- Decimal representation of ‘a’ is 97 and ‘z’ is 122
- If we convert the char datatype into int and check whether the value falls into the range, then the character will be either a vowel or a consonant. Otherwise, it’s unknown (which means there are so many possibilities).

dec	oct	hex	ch	dec	oct	hex	ch	dec	oct	hex	ch
32	40	20	(space)	64	100	40	@	96	140	60	`
33	41	21	!	65	101	41	A	97	141	61	a
34	42	22	"	66	102	42	B	98	142	62	b
35	43	23	#	67	103	43	C	99	143	63	c
36	44	24	$	68	104	44	D	100	144	64	d
37	45	25	%	69	105	45	E	101	145	65	e
38	46	26	&	70	106	46	F	102	146	66	f
39	47	27	'	71	107	47	G	103	147	67	g
40	50	28	(	72	110	48	H	104	150	68	h
41	51	29	)	73	111	49	I	105	151	69	i
42	52	2a	*	74	112	4a	J	106	152	6a	j
43	53	2b	+	75	113	4b	K	107	153	6b	k
44	54	2c	,	76	114	4c	L	108	154	6c	l
45	55	2d	-	77	115	4d	M	109	155	6d	m
46	56	2e	.	78	116	4e	N	110	156	6e	n
47	57	2f	/	79	117	4f	O	111	157	6f	o
48	60	30	0	80	120	50	P	112	160	70	p
49	61	31	1	81	121	51	Q	113	161	71	q
50	62	32	2	82	122	52	R	114	162	72	r
51	63	33	3	83	123	53	S	115	163	73	s
52	64	34	4	84	124	54	T	116	164	74	t
53	65	35	5	85	125	55	U	117	165	75	u
54	66	36	6	86	126	56	V	118	166	76	v
55	67	37	7	87	127	57	W	119	167	77	w
56	70	38	8	88	130	58	X	120	170	78	x
57	71	39	9	89	131	59	Y	121	171	79	y
58	72	3a	:	90	132	5a	Z	122	172	7a	z

Check ***vowel_consonant_nested_if_else.cpp*** on ilearn

C++ `switch` Statement

- Allow the programmer to execute a block of code among many alternatives
- Note: We can do the same thing with the if...else..if ladder. However, the syntax of the switch statement is cleaner and much easier to read and write.

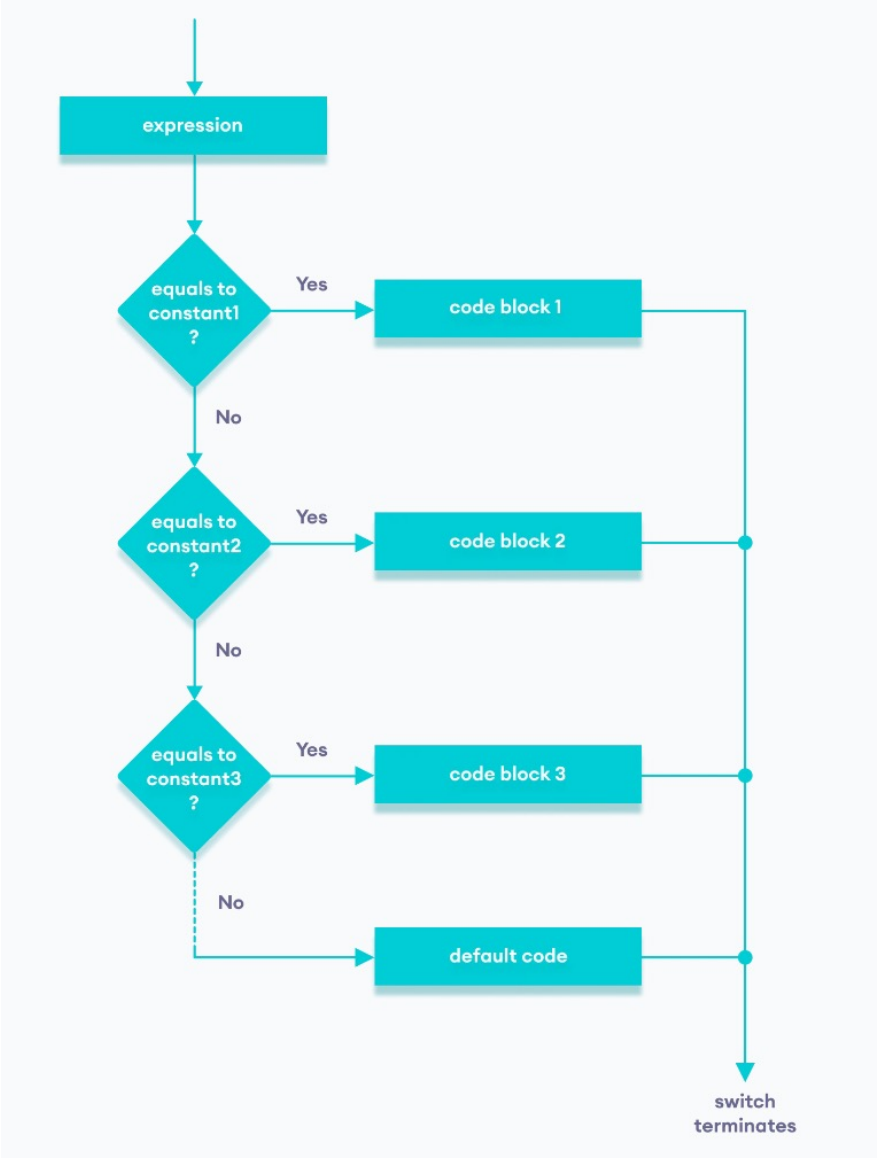


Image Source Programiz – https://www.programiz.com/cpp-programming/switch-case

The `switch` Statement Structure

```
switch (expression)
{
        case constant1:
                // Some code block
                break;
        case constant2:
                // Some code block
                break;
        …
        …
        default:
                // Code to be executed if the expression doesn't
                // match with any constant
}
```

C++ `switch` Statement – Example

A menu based program.. Check
circle_math_problems.cpp
on iLearn

Remarks

- Reference Books
  - ZyBooks, TNTech CSC 1300: Introduction to Problem Solving and Computer Programming
  - Kanetkar, Yashavant P. "Let Us C.”
  - Balagurusamy, E. “Object-Oriented Programming with C++.”