program.cpp

/* ask user for his/her age
* detect bad input – negative number
* ask user to re-enter if input is negative */

#include <iostream>
using namespace std;

int main(){
    int age;
    cout << "Please enter your age: ";
    cin >> age;
    if ( age < 0 )
    {
        cout << "Bad input. Enter again: ";
        cin >> age;
    }
    cout << "Thank you. You’re " << age
         << endl;
    return 0;
}
Repeatedly Making Decision

As we know *if* statement only make one decision.

How do we test the same decision over and over again?

- We need a loop.
While Loop Logic Flow:

- Start statement
- Condition (Test Expression)
  - True: Statement (Body of while)
  - False: Rest statement
- Rest statement
While Loop Model

```cpp
while ( condition )
{
    //statement – body of while
    //do something here
}

Ex: Repeated detect bad input
int age;
cout << "Please enter your age : ";
cin >> age;
while (age < 0)
{
    cout << "Bad input, enter again: ";
    //read in a new input
    cin >> age;
}
cout << "Thank you. You’re " << age << endl;
```
Sample While Loop

```cpp
    cin >> number;

    while (number < 0)
    {
        cout << "Bad input, enter again: ";
        cin >> number;
    }

When the above code runs, if user enters:
-1
-2
-3
0

number < 0
(-1,-2,-3,0)

read-in(-2,-3,0)
print: bad input
```

Page 6
Infinite Loop

How many times do we stay in the loop?

- As long as the condition is true
- It’s an infinite loop if condition will never be false

while (number < 0)
{
    cout << “Bad input, enter again: “;
}
- Infinite loop, testing the same value over and over again, if the number is negative, it’ll stay in the loop forever
Max limit on repeated Action

We may control the maximum times we stay in the loop.

Ex: Ask user for a positive number, give user maximum 3 tries, terminate the program after 3 strikes.

```cpp
int number, strike = 0;
cin >> number;
while (number < 0)
{
    strike = strike + 1;
    if (strike == 3) { exit(0); }
    cout << "Bad input, enter again: ";
    cin >> number;
}
```
Repeatedly Making Decision

While loop

- repeatedly detect bad input

- repetitive actions to solve a task
Ex: Print first digit
123 = 1 * 10^2 + 2 * 10^1 + 3 * 10^0
2345 = 2 * 10^3 + 3 * 10^2 + 4 * 10^1 + 5 * 10^0
what operation should we do in order to get the first digit? Divide by 100? By 1000?
- No single operation, because we don’t know how many digits are there.
- Repeatedly divide by 10, until we left with first digit.
/* Print the first digit of user input, assume user enters a positive number */

int main(){
    int num;
    cout << "Enter a positive number: " << endl;
    cin >> num;
    /* repeatedly divide num by 10 when num is greater than or equal to 10
       (at least 2 digits)
       at the end, num left with first digit */
    while (num >= 10) {
        num = num / 10;
    }
    cout << "First digit: " << num << endl;
    return 0;
}
## Arithmetic Assignment

<table>
<thead>
<tr>
<th>Operation</th>
<th>Arithmetic Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x = x + n )</td>
<td>( x += n )</td>
</tr>
<tr>
<td>( x = x - n )</td>
<td>( x -= n )</td>
</tr>
<tr>
<td>( x = x \times n )</td>
<td>( x *= n )</td>
</tr>
<tr>
<td>( x = x / n )</td>
<td>( x /= n )</td>
</tr>
<tr>
<td>( x = x % n )</td>
<td>( x %= n )</td>
</tr>
<tr>
<td>increase ( x ) by 1</td>
<td>( x++ ) or ( ++x )</td>
</tr>
<tr>
<td>decrease ( x ) by 1</td>
<td>( x-- ) or ( --x )</td>
</tr>
</tbody>
</table>

- \( n \) is any numerical values
- \( x \) is any variable

Short hand notations

---

Page 11
while loop code

/* Print the digit count of user input, assume we already declare variable num and read in a positive number */

int count = 0;

while ( num > 0 ) {
    num /= 10;
    count++;
}

cout << "Digit count: " << count << endl;