

**รายงานการทดลอง  
LAB 3-2 : Loop**

**เสนอ**  
อาจารย์กุลจรี ตันตยกุล

**จัดทำโดย**  
นายจตุภัทร์ ปานน้อย 5735512002  
Section 01  
Link Source Code : <https://github.com/gtfarng/LabIntroC>

รายงานนี้เป็นส่วนหนึ่งของรายวิชา 240-101 INTRO TO COMP PROGRAM  
ภาคการศึกษาที่ 2 ปีการศึกษา 2561  
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์  
มหาวิทยาลัยสงขลานครินทร์

**การทดลองที่ 3-2  
การทำซ้ำ Repetition หรือ Loop**

**ตอนที่ 1** โครงสร้างการทำซ้ำ for, do-while และ while

วัตถุประสงค์

* เพื่อให้นักศึกษาเข้าใจโครงสร้างของคำสั่ง for, do-while และ while
* สามารถใช้งานโครงสร้างการทำซ้ำแบบต่าง ๆ ที่ไม่ซับซ้อนได้
* สามารถเปลี่ยนการใช้งานของคำสั่ง for <==> do-while <==> while ได้

**1.1 โครงสร้างการทำซ้ำ for**

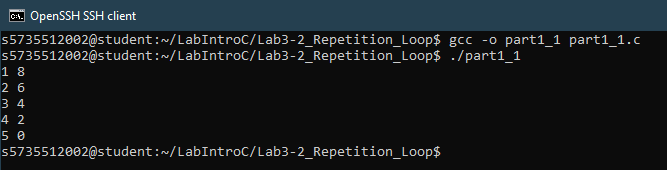
ให้ทำการพิมพ์ Source Code ดังโปรแกรมด้านล่าง

**Source code**

|  |  |
| --- | --- |
| 1. 2.  3.  4  5.  6.  7  8. 9. 10.  11.  12. | //part1\_1.c  #include<stdio.h>  int main()  {  int i,j=8;  for(i=1;i<=5;i++)  {  printf("%d %d\n",i,j);  j-=2;  }  return 0;  } |

**อธิบาย Source Code**  
บรรทัดที่ 1 comment   
บรรทัดที่ 2 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 3 ฟังก์ชันหลัก  
บรรทัดที่ 4 วงเล็บปีกกาเปิด  
บรรทัดที่ 5 ประกาศตัวแปร ชนิด Integer ชื่อ i และ ชื่อ j และกำหนดค่าเริ่มต้นให้เท่ากับ 8   
บรรทัดที่ 6 loop for โดยมีการกำหนดค่าเริ่มต้นของ i ให้เท่ากับ 1 โดยมีเงื่อนไขว่า i ต้องน้อยกว่าหรือ  
 เท่ากับ 5 แล้วทำการเพิ่มค่า i ทีละ 1   
บรรทัดที่ 7 วงเล็บปีกกาเปิดของ loop  
บรรทัดที่ 8 แสดงค่า i และ j ผ่านทางหน้าจอ  
บรรทัดที่ 9 ทำการลบค่า j ทีละ 2  
บรรทัดที่ 10 วงเล็บปีกกาปิดของ loop  
บรรทัดที่ 11 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 12 วงเล็บปีกกาปิด

**ผลการรัน**

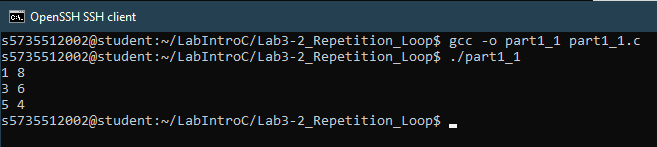


**ตอบคำถาม**

จากประโยค for(i=1;i<=5;i++)

1.ค่าเริ่มต้นของ i และ j   
Ans ค่าเริ่มต้นของ i เท่ากับ 1 และ j เท่ากับ 8  
2.จำนวนครั้งของการวนลูปกำหนดโดยตัวแปรใด และมีจำนวนครั้งเท่าใด  
Ans จำนวนครั้งของการวนลูปกำหนดโดยตัวแปร i และมีการวน loop 5 ครั้ง  
3.เงื่อนไขสิ้นสุดของการวนลูปคืออะไร  
Ans i<=5

**ผลการรัน**



**ตอบคำถาม**

จากประโยค for(i=1;i<=5;i+=2)

1.ค่าเริ่มต้นของ i และ j   
Ans ค่าเริ่มต้นของ i เท่ากับ 1 และ j เท่ากับ 8  
2.จำนวนครั้งของการวนลูปกำหนดโดยตัวแปรใด และมีจำนวนครั้งเท่าใด  
Ans จำนวนครั้งของการวนลูปกำหนดโดยตัวแปร i และมีการวน loop 3 ครั้ง  
3.เงื่อนไขสิ้นสุดของการวนลูปคืออะไร  
Ans i<=5

**1.2 โครงสร้างการทำซ้ำ do-while**

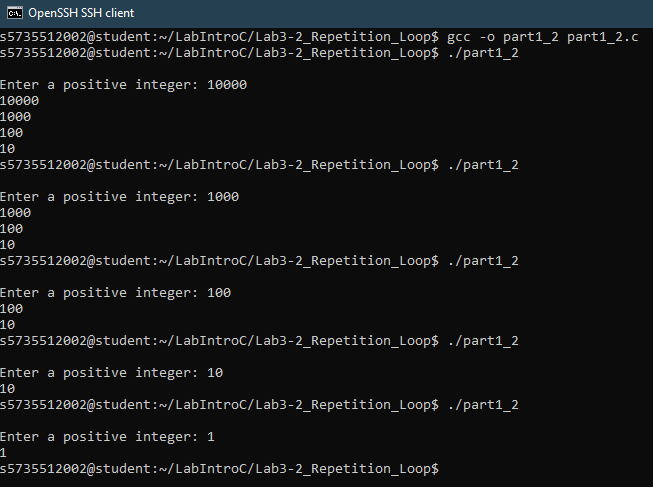
ให้ทำการพิมพ์ Source Code ดังโปรแกรมด้านล่าง

**Source code**

|  |  |
| --- | --- |
| 1. 2.  3.  4  5.  6.  7  8. 9. 10.  11.  12.  13.  14.  15. | //part1\_2.c  #include<stdio.h>  int main()  {  int num;  printf("\nEnter a positive integer: ");  scanf("%d",&num);  do  {  printf("%d \n",num);  num/=10;  }  while(num>1);  return 0;  } |

**อธิบาย Source Code**  
บรรทัดที่ 1 comment   
บรรทัดที่ 2 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 3 ฟังก์ชันหลัก  
บรรทัดที่ 4 วงเล็บปีกกาเปิด  
บรรทัดที่ 5 ประกาศตัวแปร ชนิด Integer ชื่อ num   
บรรทัดที่ 6 แสดงคำว่า “Enter a positive integer: ”ผ่านทางหน้าจอ  
บรรทัดที่ 7 เก็บค่า integer ไว้ที่ตัวแปร num  
บรรทัดที่ 8 การทำงานของ do  
บรรทัดที่ 9 วงเล็บปีกกาเปิดของ do  
บรรทัดที่ 10 แสดงค่า num ผ่านทางหน้าจอ  
บรรทัดที่ 11 ทำการหารค่า num ทีละ 10 (num=num/10;)   
บรรทัดที่ 12 วงเล็บปีกกาปิดของ do  
บรรทัดที่ 13 เช็คเงื่อนไขโดยที่ num มากกว่า 1  
บรรทัดที่ 14 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 15 วงเล็บปีกกาปิด

**ผลการรัน**



**ตอบคำถาม**

1.จากโปรแกรม part1\_2.c จงระบุว่าค่าที่ถูกพิมพ์ทางหน้าจอคือค่าอะไร เมื่อใส่ input เป็น 10000, 1000, 10 และ 1  
Ans

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input** | **10000** | **1000** | **10** | **1** |
| **Output** | 10000  1000  100  10 | 1000  100  10 | 10 | 1 |

2.ตัวแปรอะไรใช้ในการกำหนดเงื่อนไขในการตัดสินใจให้ออกจากการทำซ้ำ   
Ans ตัวแปร num  
3.ตัวแปรที่ใช้กำหนดเงื่อนไขในการวนซ้ำมีการเปลี่ยนแปลงค่าอย่างไร และมีการกำหนดค่าเริ่มต้นอย่างไร  
Ans ตัวแปร num มีการหารค่าทีละ 10 และค่าเริ่มต้นก็ได้จาก input ที่รับเข้ามา  
4.ถ้ามีการแก้ไขโปรแกรมดังกล่าวให้เป็นโปรแกรมตามด้านล่าง ผลลัพธ์ที่เกิดขึ้นเป็นอย่างไรและเหตุผล  
Ans เมื่อโปรแกรมเริ่มทำงานโปรแกรมจะแสดงค่าที่ทำการป้อนครั้งแรกก่อน ถึงจะทำการเช็คเงื่อนไขแล้วจึงดำเนินการการทำงานขั้นต่อไป ส่วนโปรแกรมด้านล่างจะเริ่มการทำงานโดยทำการเช็คเงื่อนไขก่อนที่จะดำเนินการการทำงานขั้นต่อไป

**1.3 โครงสร้างการทำซ้ำ while**

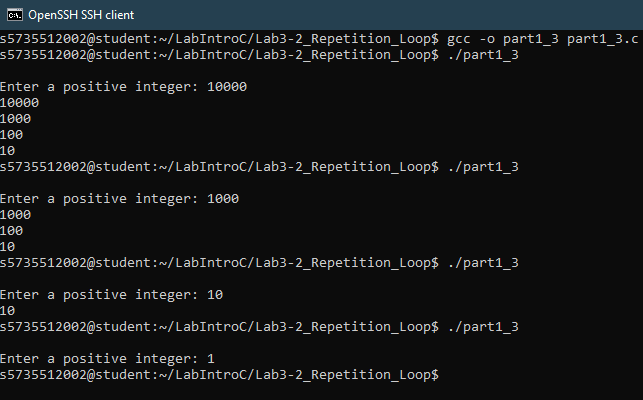
ให้ทำการพิมพ์ Source Code ดังโปรแกรมด้านล่าง

**Source code**

|  |  |
| --- | --- |
| 1. 2.  3.  4  5.  6.  7  8. 9. 10.  11.  12.  13.  14. | //part1\_3.c  #include<stdio.h>  int main()  {  int num;  printf("\nEnter a positive integer: ");  scanf("%d",&num);  while(num>1)  {  printf("%d \n",num);  num/=10;  }  return 0;  } |

**อธิบาย Source Code**  
บรรทัดที่ 1 comment   
บรรทัดที่ 2 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 3 ฟังก์ชันหลัก  
บรรทัดที่ 4 วงเล็บปีกกาเปิด  
บรรทัดที่ 5 ประกาศตัวแปร ชนิด Integer ชื่อ num   
บรรทัดที่ 6 แสดงคำว่า “Enter a positive integer: ”ผ่านทางหน้าจอ  
บรรทัดที่ 7 เก็บค่า integer ไว้ที่ตัวแปร num  
บรรทัดที่ 8 เช็คเงื่อนไขโดยที่ num มากกว่า 1  
บรรทัดที่ 9 วงเล็บปีกกาเปิดของ while  
บรรทัดที่ 10 แสดงค่า num ผ่านทางหน้าจอ  
บรรทัดที่ 11 ทำการหารค่า num ทีละ 10 (num=num/10;)   
บรรทัดที่ 12 วงเล็บปีกกาปิดของ while  
บรรทัดที่ 13 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 14 วงเล็บปีกกาปิด

**ผลการรัน**



**ตอบคำถาม**

1.จากโปรแกรม part1\_2.c จงระบุว่าค่าที่ถูกพิมพ์ทางหน้าจอคือค่าอะไร เมื่อใส่ input เป็น 10000, 1000, 10 และ 1  
Ans

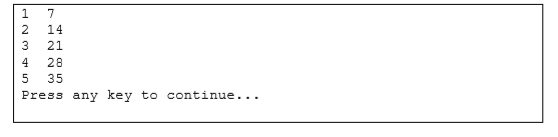
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input** | **10000** | **1000** | **10** | **1** |
| **Output** | 10000  1000  100  10 | 1000  100  10 | 10 |  |

2.จงบอกความแตกต่างระหว่างโปรแกรม part1\_2 และ part1\_3 ว่ามีการสดงผลลัพธ์ต่างกันอย่างไร เพราะเหตุใดเมื่อป้อนค่า 10 และ 1  
Ans

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **loop** | **while** | | **Do-while** | |
| **Input** | **10** | **1** | **10** | **1** |
| **Output** | 10 |  | 10 | 1 |

loop while จะเริ่มต้นการทำงานโดยการเช็คเงื่อนไขก่อนถึงจะเริ่มการทำงาน ส่วน loop do-while จะเริ่มต้นการทำงาน โดยเริ่มทำงานก่อน 1 ครั้งจากนั้นทำการเช็คเงื่อนไขแล้วจึงจะเริ่มการทำงานต่อไป

**Checkpoint 1** จงเขียนโปรแกรมเพื่อให้ได้ผลลัพธ์ด้านล่าง โดยใช้คำสั่งวนลูปใดก็ได้

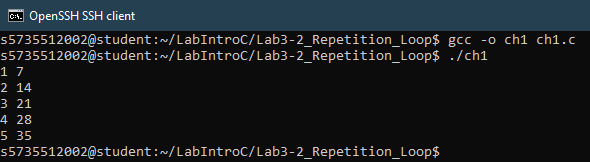


**Source code**

|  |  |
| --- | --- |
| 1. 2.  3.  4  5.  6.  7  8. 9. 10.  11.  12. | #include<stdio.h>  int main()  {  int i=0;  do  {  printf("%d %d\n",i+1,7\*(i+1));  i++;  }  while(i<5);  return 0;  } |

**อธิบาย Source Code**  
บรรทัดที่ 1 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 2 ฟังก์ชันหลัก  
บรรทัดที่ 3 วงเล็บปีกกาเปิด  
บรรทัดที่ 4 ประกาศตัวแปร ชนิด Integer ชื่อ i กำหนดค่าเริ่มต้นให้เท่ากับ 0  
บรรทัดที่ 5 การทำงานของ do  
บรรทัดที่ 6 วงเล็บปีกกาเปิดของ do  
บรรทัดที่ 7 แสดงค่าจาก i+1 และ 7\*(i+1)) ผ่านทางหน้าจอ  
บรรทัดที่ 8 เพิ่มค่า i ทีละ 1  
บรรทัดที่ 9 วงเล็บปีกกาปิดของ do  
บรรทัดที่ 10 เช็คเงื่อนไขโดยที่ i น้อยกว่า 5  
บรรทัดที่ 11 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 12 วงเล็บปีกกาปิด

**ผลการรัน**



**การทดลองที่ 2.1 โปรแกรมดาวกระชากใจ**

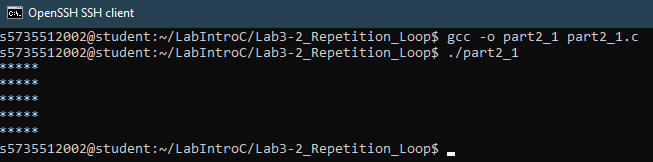
ให้ทำการพิมพ์ Source Code ดังโปรแกรมด้านล่าง

**Source code**

|  |  |
| --- | --- |
| 1. 2.  3.  4  5.  6.  7  8. 9. 10.  11.  12.  13.  14.  15.  16. | //part2\_1.c  #include<stdio.h>  #define MAX 5  int main()  {  int i,j;  for(i=MAX;i>0;i--)  {  for(j=0;j<MAX;j++)  {  printf("\*");  }  printf("\n");  }  return 0;  } |

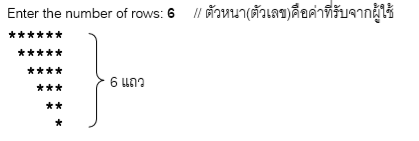
**อธิบาย Source Code**  
บรรทัดที่ 1 comment   
บรรทัดที่ 2 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 3 ฟังก์ชันหลัก  
บรรทัดที่ 4 วงเล็บปีกกาเปิด  
บรรทัดที่ 5 ประกาศตัวแปร ชนิด Integer ชื่อ i และ ชื่อ j   
บรรทัดที่ 6   
บรรทัดที่ 7 วงเล็บปีกกาเปิดของ loop  
บรรทัดที่ 8 แสดงค่า i และ j ผ่านทางหน้าจอ  
บรรทัดที่ 9 ทำการลบค่า j ทีละ 2  
บรรทัดที่ 10 วงเล็บปีกกาปิดของ loop  
บรรทัดที่ 11   
บรรทัดที่ 12   
บรรทัดที่ 13   
บรรทัดที่ 14   
บรรทัดที่ 15 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 16 วงเล็บปีกกาปิด

**ผลการรัน**



**สรุปผลการทดลอง**

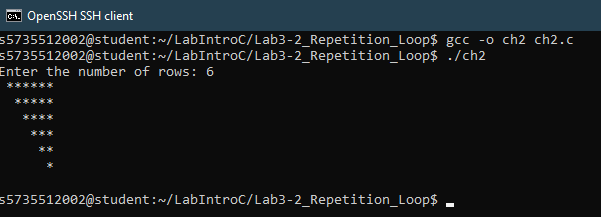
**Checkpoint 2** จงเขียนโปรแกรมเพื่อให้ได้ผลลัพธ์ด้านล่าง เมื่อตัวเลข(ตัวหนา)ป้อนมาจากผู้ใช้



**Source code**

|  |  |
| --- | --- |
| 1. 2.  3.  4  5.  6.  7  8. 9. 10.  11.  12.  13.  14.  15.  16.  17.  18.  19. | #include<stdio.h>  int main()  {  int n,i,j;  printf("Enter the number of rows: ");  scanf("%d",&n);  for(i=0;i<=n;i++)  {  for(j=0;j<=n;j++)  {  if(i<j)  printf("\*");  else  printf(" ");  }  printf("\n");  }  return 0;  } |

**อธิบาย Source Code**  
บรรทัดที่ 1 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 2 ฟังก์ชันหลัก  
บรรทัดที่ 3 วงเล็บปีกกาเปิด  
บรรทัดที่ 4   
บรรทัดที่ 5   
บรรทัดที่ 6   
บรรทัดที่ 7   
บรรทัดที่ 8   
บรรทัดที่ 9   
บรรทัดที่ 10   
บรรทัดที่ 11   
บรรทัดที่ 12   
บรรทัดที่ 13   
บรรทัดที่ 14   
บรรทัดที่ 15   
บรรทัดที่ 16   
บรรทัดที่ 17   
บรรทัดที่ 18 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 19 วงเล็บปีกกาปิด  
  
**ผลการรัน**



**สรุปผลการทดลอง**

**งานท้ายการทดลอง**

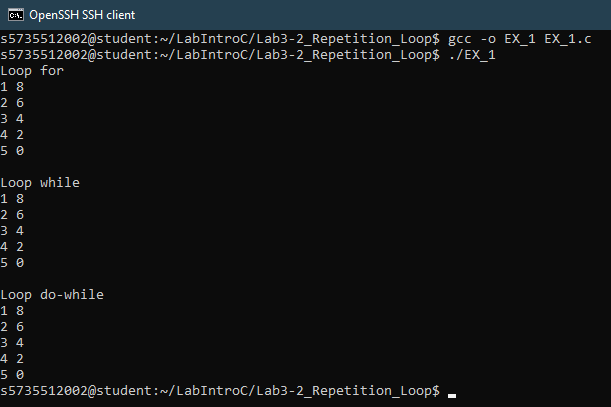
1.จากการทดลองตอนที่ 1.1 จงเขียนโปรแกรมโดยใช้โครงสร้างแบบ do-while และ while โดยที่ยังคงให้ผลลัพธ์เหมือนเดิม

**Source code**

|  |  |
| --- | --- |
| 1. 2.  3.  4  5.  6.  7  8. 9. 10.  11.  12.  13.  14.  15.  16.  17.  18.  19.  20.  21.  22.  23.  24.  25.  26.  27.  28.  29. | #include<stdio.h>  int main()  {  int i=1,j=8;  printf("Loop for\n");  for(i=1;i<=5;i++)  {  printf("%d %d\n",i,j);  j-=2;  }  i=1;j=8;  printf("\nLoop while\n");  while(i<=5)  {  printf("%d %d\n",i,j);  j-=2;  i++;  }  i=1;j=8;  printf("\nLoop do-while\n");  do  {  printf("%d %d\n",i,j);  j-=2;  i++;  }  while(i<=5);  return 0;  } |

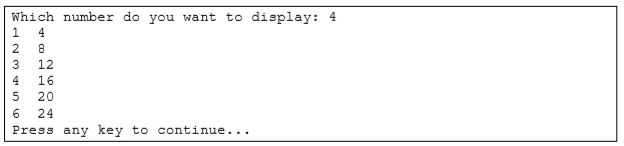
**อธิบาย Source Code**  
บรรทัดที่ 1 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 2 ฟังก์ชันหลัก  
บรรทัดที่ 3 วงเล็บปีกกาเปิด  
บรรทัดที่ 4   
บรรทัดที่ 5   
บรรทัดที่ 6   
บรรทัดที่ 7   
บรรทัดที่ 8   
บรรทัดที่ 9   
บรรทัดที่ 10   
บรรทัดที่ 11   
บรรทัดที่ 12   
บรรทัดที่ 13   
บรรทัดที่ 14   
บรรทัดที่ 15   
บรรทัดที่ 16   
บรรทัดที่ 17   
บรรทัดที่ 18 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 19 วงเล็บปีกกาปิด  
บรรทัดที่ 20 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 21 ฟังก์ชันหลัก  
บรรทัดที่ 22 วงเล็บปีกกาเปิด  
บรรทัดที่ 23   
บรรทัดที่ 24   
บรรทัดที่ 25   
บรรทัดที่ 26   
บรรทัดที่ 27   
บรรทัดที่ 28   
บรรทัดที่ 29

**ผลการรัน**



**สรุปผลการทดลอง**

2.จากตัวอย่างผู้ใช้ป้อนค่า 4 แล้วโปรแกรมแสดงตารางสูตรคูณของตัวเลข 4 แต่มีเงื่อนไขคือ ให้แสดงค่าผลคูณ จาก 1 ถึง 6 แต่ค่าที่ต้องแสดงต้องมีค่าไม่เกินเลข 60 ( เช่น ถ้าใส่ 13 จะแสดงได้ถึงแค่ 13x4 คือ 52 )

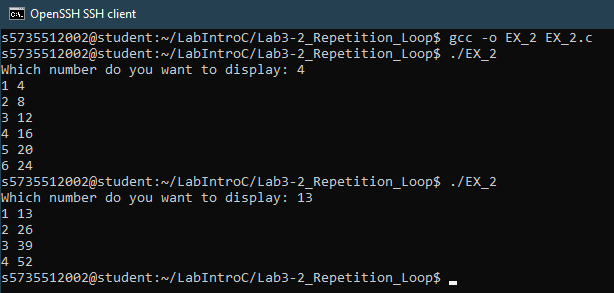


**Source code**

|  |  |
| --- | --- |
| 1. 2.  3.  4  5.  6.  7  8. 9. 10.  11.  12.  13.  14. | #include<stdio.h>  int main()  {  int i=0,n;  printf("Which number do you want to display: ");  scanf("%d",&n);  do  {  printf("%d %d\n",i+1,n\*(i+1));  i++;  }  while(n\*(i+1)<60&&i<6);  return 0;  } |

**อธิบาย Source Code**  
บรรทัดที่ 1 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 2 ฟังก์ชันหลัก  
บรรทัดที่ 3 วงเล็บปีกกาเปิด  
บรรทัดที่ 4   
บรรทัดที่ 5   
บรรทัดที่ 6   
บรรทัดที่ 7   
บรรทัดที่ 8   
บรรทัดที่ 9   
บรรทัดที่ 10   
บรรทัดที่ 11   
บรรทัดที่ 12   
บรรทัดที่ 13 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 14 วงเล็บปีกกาปิด

**ผลการรัน**



**สรุปผลการทดลอง**

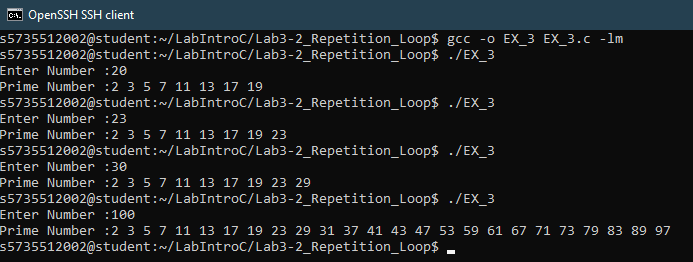
3.จงเขียนโปรแกรม เพื่อแสดงเลขจำนวนเฉพาะ (prime numbers) ที่มีค่าน้อยกว่าหรือเท่ากับ n (n>1) โดยรับค่า n จากผู้ใช้ เช่น เมื่อผู้ใช้ป้อนค่า 20



**Source code**

|  |  |
| --- | --- |
| 1. 2.  3.  4  5.  6.  7  8. 9. 10.  11.  12.  13.  14.  15.  16.  17.  18.  19. 20. 21. | #include<stdio.h>  #include<Math.h>  int main()  {  int n,i,j,a,b;  printf("Enter Number :");  scanf("%d",&n);  printf("Prime Number :");  for(i=2;i<=n;i++)  {  a = sqrt(i);  b = 1;  for(j=2;j<=a;j++)  {  b = b&&!(i%j==0);  }  if(b==1)  printf("%d ",i);  }  return 0;  } |

**อธิบาย Source Code**  
บรรทัดที่ 1 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 2 ฟังก์ชันหลัก  
บรรทัดที่ 3 วงเล็บปีกกาเปิด  
บรรทัดที่ 4   
บรรทัดที่ 5   
บรรทัดที่ 6   
บรรทัดที่ 7   
บรรทัดที่ 8   
บรรทัดที่ 9   
บรรทัดที่ 10   
บรรทัดที่ 11   
บรรทัดที่ 12   
บรรทัดที่ 13   
บรรทัดที่ 14   
บรรทัดที่ 15   
บรรทัดที่ 16   
บรรทัดที่ 17   
บรรทัดที่ 18   
บรรทัดที่ 19   
บรรทัดที่ 20 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 21 วงเล็บปีกกาปิด  
  
**ผลการรัน**



**สรุปผลการทดลอง**