

**รายงานการทดลอง  
LAB 4-2 : Arrays 2D**

**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 4-2   
อาร์เรย์ Arrays (ตอนที่ 2 อาร์เรย์สองมิติและหลายมิติ)**

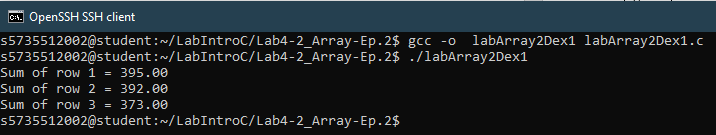
**วัตถุประสงค์**  
 เพื่อให้นักศึกษาเข้าใจและสามารถใช้งานอาร์เรย์แบบ 2 มิติและหลายมิติได้

**การทดลอง อาร์เรย์ 2 มิติ**   
 ให้ทำการพิมพ์ Source Code ดังโปรแกรมด้านล่าง

**Source code**

|  |  |
| --- | --- |
| 1. 2.  3.  4  5.  6.  7  8. 9. 10.  11.  12.  13.  14.  15. | //labArray2Dex1.c  #include<stdio.h>  int main()  { int i, j, scores[3][5] = { {84, 71, 96, 65, 79},  {90, 55, 83, 68, 96},  {61, 77, 82, 94, 59}};  float sum;  for (i=0; i<3; i++)  {  for (j=0, sum=0.0; j<5; j++)  sum = sum+scores[i][j];  printf("Sum of row %d = %.2f \n",i+1, sum);  }  return 0;  } |

**ผลการรัน**



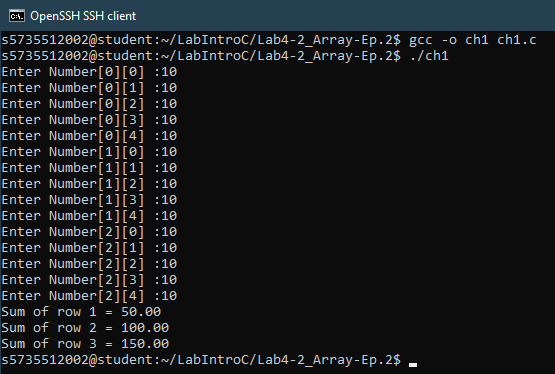
**Checkpoint 1**

* จงปรับปรุงโปรแกรม เพื่อให้สามารถรับค่าข้อมูลจำนวนเต็มไปเก็บไว้ในอาร์เรย์ scores โดยใช้คำสั่ง scanf แทนการกำหนดค่าเริ่มต้น โดยให้ผลของโปรแกรมเหมือนเดิม
* เพิ่มการคำนวณ หาค่าเฉลี่ยรวม ของทุกจำนวนในอาร์เรย์และแสดงผลลัพธ์ก่อนจบโปรแกรม

**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. | #include<stdio.h>  int main()  {  int i, j, scores[3][5];  float sum=0;  for(i=0;i<3;i++)  {  for(j=0;j<5;j++)  {  printf("Enter Number[%d][%d] :",i,j);  scanf("%d",&scores[i][j]);  }  }  for (i=0; i<3; i++)  {  for (j=0; j<5; j++)  {  sum = sum+scores[i][j];  }  printf("Sum of row %d = %.2f\n",i+1,sum);  }  return 0;  } |

**ผลการรัน**



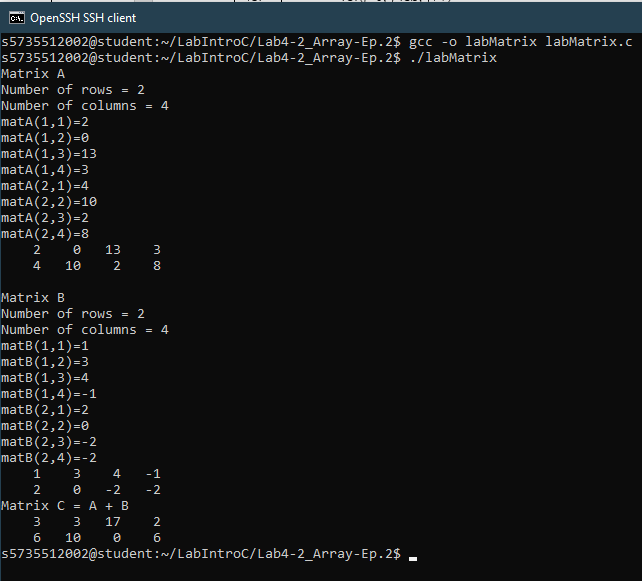
**การทดลอง เมตริกส์ (Maxtrix)**

ให้ทำการพิมพ์ Source Code ดังโปรแกรมด้านล่าง

**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.  30.  31.  32.  33.  34.  35.  36.  37.  38.  39.  40.  41.  42.  43.  44.  45.  46.  47.  48.  49.  50.  51.  52.  53.  54.  55.  56.  57.  58.  59.  60.  61.  62.  63.  64.  65.  66.  67. | //labMatrix.c  #include <stdio.h>  #define MAX 10  int main()  {  int matA[MAX][MAX];  int matB[MAX][MAX];  int matC[MAX][MAX];  int ra,rb,ca,cb,i,j;  printf("Matrix A\n");  printf("Number of rows = ");  scanf("%d",&ra);  printf("Number of columns = ");  scanf("%d",&ca);  // Input elements of Matrix A(ra x ca)  for (i=0; i<ra; i++)  {  for(j=0; j<ca; j++)  {  printf("matA(%d,%d)=",i+1,j+1);  scanf("%d",&matA[i][j]);  }  }  // Show elements of Matrix A(ra x ca)  for (i=0; i<ra; i++)  {  for(j=0; j<ca; j++)  printf("%5d",matA[i][j]);  printf("\n");  }  printf("\nMatrix B\n");  printf("Number of rows = ");  scanf("%d",&rb);  printf("Number of columns = ");  scanf("%d",&cb);  // Input elements of Matrix B(rb x cb)  for (i=0; i<rb; i++)  {  for(j=0; j<cb; j++)  {  printf("matB(%d,%d)=",i+1,j+1);  scanf("%d",&matB[i][j]);  }  }  // Show elements of Matrix B(rb x cb)  for (i=0; i<rb; i++)  {  for(j=0; j<cb; j++)  printf("%5d",matB[i][j]);  printf("\n");  }  printf("Matrix C = A + B \n");  // Addition: Matrix C = A + B  for (i=0; i<ra; i++)  {  for(j=0; j<ca; j++)  matC[i][j]=matA[i][j]+matB[i][j];  }  // Show elements of Matrix C(ra x ca)  for (i=0; i<ra; i++)  {  for(j=0; j<ca; j++)  printf("%5d",matC[i][j]);  printf("\n");  }  return 0;  } |

**ผลการรัน**



-เมตริกส์ C มีค่าเท่าไร ? มีขนาดเท่าไร ?  
Ans เมตริกส์ C มีค่า และมีขนาด 4X2

-ขนาดของเมตริกส์ A และ B กำหนดโดยตัวแปรใดบ้าง ?  
Ans ขนาดของเมตริกส์ A กำหนดโดยตัวแปร ra และ ca   
 ส่วนขนาดของเมตริกส์ B กำหนดโดยตัวแปร rb และ cb

-จากโปรแกรม เราสามารถกำหนดขนาดของเมตริกส์ได้สูงสุดเท่าไร ? จะแก้ไขอย่างไร ?  
Ans จากโปรแกรมเราสามารถกำหนดขนาดของเมตริกส์ได้สูงสุด คือ 10X10   
 และเราสามารถแก้ไขขนาดเมตริกส์สูงสุดได้ที่ #define MAX 10

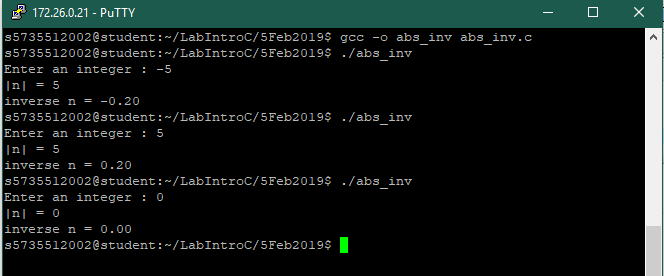
**ตอนที่ 2** ตัวดำเนินการเงื่อนไข (Conditional operator) ? :  
 ให้ทำการพิมพ์ Source Code ดังโปรแกรมด้านล่าง

**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. | // abs\_inv.c : absolute and inverse of integer  #include <stdio.h>  #include <stdlib.h>  int main ()  {  int n, abs\_n;  float inv\_n;  printf("Enter an integer : ");  scanf("%d",&n);  /\* abs\_n = n<0 ? -n : n;    //conditional operator  inv\_n = n!=0 ? 1.0/n : 0;  \*/ //conditional operator  if(n<0)  abs\_n=-n;  else  abs\_n=n;  if(n!=0)  inv\_n=1.0/n;  else  inv\_n=0;  printf("|n| = %d\n",abs\_n);  printf("inverse n = %.2f\n",inv\_n);  return 0;  } |

**อธิบาย Source Code**  
บรรทัดที่ 1 comment   
บรรทัดที่ 2 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 3 เรียกใช้ library Standard General utilities  
บรรทัดที่ 4 ฟังก์ชันหลัก  
บรรทัดที่ 5 วงเล็บปีกกาเปิด  
บรรทัดที่ 6 ประกาศตัวแปรชนิด Integer ชื่อ n และชื่อ abs\_n  
บรรทัดที่ 7 ประกาศตัวแปรชนิด float ชื่อ inv\_n   
บรรทัดที่ 8 แสดงคำว่า "Enter an integer : " ผ่านทางหน้าจอ  
บรรทัดที่ 9 ทำการรับค่าเป็น integer เก็บไว้ที่ n  
บรรทัดที่ 10 เงื่อนไขแบบ Conditional operator ของ abs\_n  
บรรทัดที่ 11   
บรรทัดที่ 12 comment   
บรรทัดที่ 13 เงื่อนไขแบบ Conditional operator ของ inv\_n  
บรรทัดที่ 14 comment   
บรรทัดที่ 15 เงื่อนไขแบบ if-else ของ abs\_n กรณี if ถ้า n น้อยกว่า 0  
บรรทัดที่ 16 ให้ abs\_n=-n;  
บรรทัดที่ 17 กรณี else  
บรรทัดที่ 18 ให้ abs\_n=n;  
บรรทัดที่ 19 เงื่อนไขแบบ if-else ของ inv\_n กรณี if ถ้า n ไม่เท่ากับ 0  
บรรทัดที่ 20 ให้ abs\_n=-n;  
บรรทัดที่ 21 กรณี else  
บรรทัดที่ 22 ให้ abs\_n=n;  
บรรทัดที่ 23 แสดงคำว่า "|n| = %d" ผ่านทางหน้าจอ  
บรรทัดที่ 24 แสดงคำว่า "inverse n = %.2f" ผ่านทางหน้าจอ  
บรรทัดที่ 25 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 26 วงเล็บปีกกาปิด

**ผลการรัน**



**สรุปผลการทดลอง**

จากโปรแกรมดังกล่าว โปรแกรมจะให้ทำการรับค่าจำนวนมาหนึ่ง โดยโปรแกรมนี้เป็นโปรแกรมหาค่าค่าสัมบูรณ์หรือ Absolute value และค่า Inverse การหาค่าสัมบูรณ์หรือ Absolute value มีเงื่อนไขว่า ถ้า n หรือค่าที่ป้อนเข้ามาน้อยกว่า 0 จะเข้ากรณี if โดยการคูณด้วยลบหนึ่ง ส่วนกรณี else จะคงค่าเดิมเอาไว้ ส่วนการหาค่า Inverse มีเงื่อนไขว่า ถ้า n ไม่เท่ากับ 0 จะเข้ากรณี if โดยการเอาหนึ่งหารด้วย n หรือค่าที่ป้อน ส่วนกรณี else จะกำหนดค่าให้เท่ากับศูนย์

//conditional operator

//หาค่าสัมบูรณ์หรือ Absolute value

abs\_n = n<0 ? -n : n;

//หาค่า Inverse

inv\_n = n!=0 ? 1.0/n : 0;

//if-else

//หาค่าสัมบูรณ์หรือ Absolute value

if(n<0)   
 abs\_n=-n;

else   
 abs\_n=n;

//หาค่า Inverse

if(n!=0)

inv\_n=1.0/n;

else

inv\_n=0;

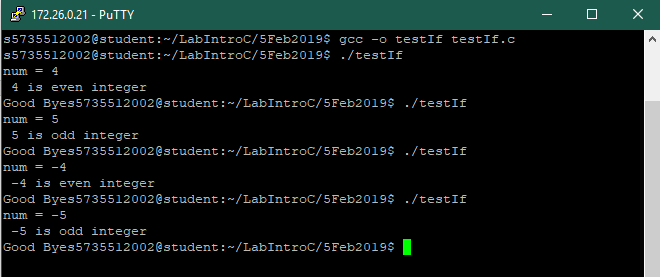
**ตอนที่ 3** การทำงานโดยใช้คำสั่ง if-else  
 ให้ทำการพิมพ์ Source Code ดังโปรแกรมด้านล่าง

**Source code**

|  |  |
| --- | --- |
| 1. 2.  3.  4  5.  6.  7  8. 9. 10.  11.  12.  13. | #include<stdio.h>  int main()  {  int num;  printf("num = ");  scanf("%d", &num);  if (num%2 == 0)  printf(" %d is even integer \n", num);  else  printf(" %d is odd integer \n", num);  printf("Good Bye");  return 0;  } |

**อธิบาย Source Code**  
บรรทัดที่ 1 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 2 ฟังก์ชันหลัก  
บรรทัดที่ 3 วงเล็บปีกกาเปิด  
บรรทัดที่ 4 ประกาศตัวแปรชนิด Integer ชื่อ num   
บรรทัดที่ 5 แสดงคำว่า "num =" ผ่านทางหน้าจอ  
บรรทัดที่ 6 ทำการรับค่าเป็น integer เก็บไว้ที่ num  
บรรทัดที่ 7 เงื่อนไขแบบ if-else กรณี if เช็คเงื่อนไขว่า num เท่ากับ 0 หรือไม่ ถ้าเป็นจริง  
บรรทัดที่ 8 แสดงคำว่า "%d is even integer " ผ่านทางหน้าจอ  
บรรทัดที่ 9 กรณี else  
บรรทัดที่ 10 แสดงคำว่า "%d is odd integer " ผ่านทางหน้าจอ  
บรรทัดที่ 11 แสดงคำว่า "Good Bye" ผ่านทางหน้าจอ  
บรรทัดที่ 12 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 13 วงเล็บปีกกาปิด

**ผลการรัน**



**สรุปผลการทดลอง**

จากการทดลองดังกล่าว โปรแกรมนี้เป็นโปรแกรมเช็คจำนวนที่ทำการป้อนเข้าไปว่าเป็นเลขคู่ (even number) หรือเลขคี่ (odd number) โดยการใช้เงื่อนไขแบบ if-else โดยมีเงื่อนไขว่า ในกรณี if เช็คเงื่อนไขว่า num เท่ากับ 0 หรือไม่ ถ้าเป็นจริงให้แสดงข้อความว่า เป็นเลขคู่ (even number) ถ้าเป็นเท็จให้แสดงข้อความว่า เลขคี่ (odd number)

**Checkpoint 1**

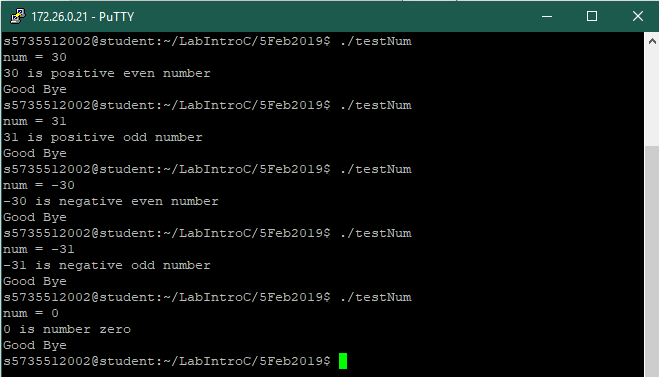
จงเขียนโปรแกรม testNum.c เพื่อตรวจสอบว่า ตัวเลขจำนวนเต็มเป็นเลขคู่ (Even) หรือเลขคี่ (Odd) และเป็นจำนวนเต็มบวก (positive) หรือจำนวนเต็มลบ (negative) หรือมีค่าเป็นศูนย์ โดยสามารถแสดงผลลัพธ์ ซึ่งแบ่งออกเป็น 5 กรณี คือ positive even number, positive odd number, zero และ negative even

**Source code**

|  |  |
| --- | --- |
| 1. 2.  3.  4  5.  6.  7  8. 9. 10.  11.  12.  13.  14.  15.  16.  17.  18.  19.  20. | // testNum.c  #include<stdio.h>  int main()  {  int num;  printf("num = ");  scanf("%d", &num);  if (num%2 == 0 && num<0)  printf("%d is negative even number\n",num);  else if(num%2 == 0 && num >= 1)  printf("%d is positive even number\n",num);  else if(num%2 == -1 && num< 0)  printf("%d is negative odd number\n",num);  else if(num%2 ==1 && num>=0)  printf("%d is positive odd number\n",num);  else if(num==0)  printf("%d is number zero\n",num);  printf("Good Bye\n");  return 0;  } |

**อธิบาย Source Code**  
บรรทัดที่ 1 Comment  
บรรทัดที่ 2 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 3 ฟังก์ชันหลัก  
บรรทัดที่ 4 วงเล็บปีกกาเปิด  
บรรทัดที่ 5 ประกาศตัวแปรชนิด Integer ชื่อ num  
บรรทัดที่ 6 แสดงคำว่า "num =" ผ่านทางหน้าจอ  
บรรทัดที่ 7 ทำการรับค่าเป็น integer เก็บไว้ที่ num  
บรรทัดที่ 8 เงื่อนไขแบบ if-else กรณี if เช็คเงื่อนไขว่า num%2 เท่ากับ 0 หรือไม่และ num น้อยกว่า 0 บรรทัดที่ 9 ถ้าเป็นจริงให้แสดงคำว่า "%d is negative even number" ผ่านทางหน้าจอ  
บรรทัดที่ 10 กรณี else if เช็คเงื่อนไขว่า num%2 เท่ากับ -1 หรือไม่และ num มากกว่าหรือเท่ากับ 0  
บรรทัดที่ 11 ถ้าเป็นจริงให้แสดงคำว่า "%d is positive even number " ผ่านทางหน้าจอ  
บรรทัดที่ 12 กรณี else if เช็คเงื่อนไขว่า num%2 เท่ากับ 1 หรือไม่และ num มากกว่าหรือเท่ากับ 0  
บรรทัดที่ 13 ถ้าเป็นจริงให้แสดงคำว่า "%d is negative odd number " ผ่านทางหน้าจอ  
บรรทัดที่ 14 กรณี else if เช็คเงื่อนไขว่า num%2 เท่ากับ 0 หรือไม่และ num มากกว่าหรือเท่ากับ 0  
บรรทัดที่ 15 ถ้าเป็นจริงให้แสดงคำว่า "%d is positive odd number " ผ่านทางหน้าจอ  
บรรทัดที่ 16 กรณี else if เช็คเงื่อนไขว่า num เท่ากับ 0 หรือไม่   
บรรทัดที่ 17 ถ้าเป็นจริงให้แสดงคำว่า "%d is number zero " ผ่านทางหน้าจอ  
บรรทัดที่ 18 แสดงคำว่า "Good Bye" ผ่านทางหน้าจอ  
บรรทัดที่ 19 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 20 วงเล็บปีกกาปิด

**ผลการรัน**



**สรุปผลการทดลอง**

จากโปรแกรมดังกล่าว เป็นโปรแกรมเช็คจำนวนที่ทำการป้อนเข้าไปว่าเป็นจำนวนเต็มบวกซึ่งเป็นจำนวนคู่ (positive even number) หรือเป็นจำนวนเต็มบวกซึ่งเป็นจำนวนคี่ (positive odd number) หรือเป็นจำนวนเต็มลบซึ่งเป็นจำนวนคู่ (negative even number) หรือเป็นจำนวนเต็มลบซึ่งเป็นจำนวนคี่ (negative odd number) หรือเป็นจำนวนเต็มศูนย์ (zero number) โดยมีเงื่อนไขทางเลือกอยู่ 5 กรณี

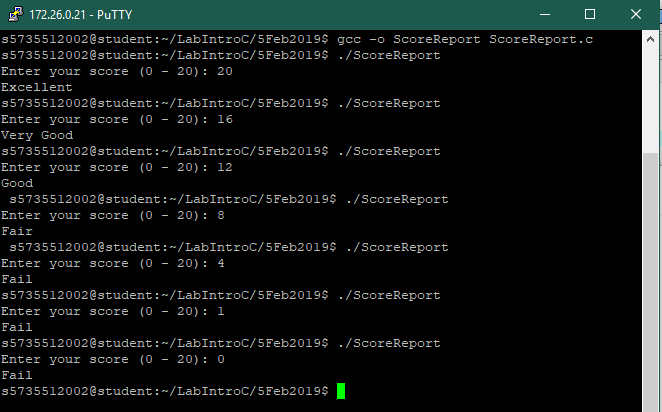
**ตอนที่ 4** โครงสร้างแบบทางเลือก ด้วยการใช้คำสั่ง switch-case

ให้ทำการพิมพ์ Source Code ดังโปรแกรมด้านล่าง **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.  30.  31.  32.  33.  34.  35. | //scoreReport.c  #include<stdio.h>  int main()  {  int score;  double grade;  printf("Enter your score (0 - 20): ");  scanf("%d", &score);  grade = score/4;  switch((int)grade)  {  case 5:  printf("Excellent\n");  break;  case 4:  printf("Very Good \n");  break;  case 3:  printf("Good \n ");  break;  case 2:  printf("Fair \n ");  break;  case 1:  printf("Fail\n");  break;  case 0:  printf("Fail\n");  break;  default:  printf("Out of range\n");  break;  }  return 0;  } |

**อธิบาย Source Code**  
บรรทัดที่ 1 Comment  
บรรทัดที่ 2 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 3 ฟังก์ชันหลัก  
บรรทัดที่ 4 วงเล็บปีกกาเปิด  
บรรทัดที่ 5 ประกาศตัวแปรชนิด Integer ชื่อ score  
บรรทัดที่ 6 ประกาศตัวแปรชนิด double ชื่อ grade  
บรรทัดที่ 7 แสดงคำว่า " Enter your score (0 - 20): " ผ่านทางหน้าจอ  
บรรทัดที่ 8 ทำการรับค่าเป็น integer เก็บไว้ที่ score  
บรรทัดที่ 9 grade = score/4;  
บรรทัดที่ 10 การทำงานแบบ switch-case  
บรรทัดที่ 11 วงเล็บปีกกาปิด ของ switch-case  
บรรทัดที่ 12 case เท่ากับ 5  
บรรทัดที่ 13 แสดงคำว่า " Excellent " ผ่านทางหน้าจอ  
บรรทัดที่ 14 หยุดการทำงาน  
บรรทัดที่ 15 case เท่ากับ 4  
บรรทัดที่ 16 แสดงคำว่า " Very Good " ผ่านทางหน้าจอ  
บรรทัดที่ 17 หยุดการทำงาน  
บรรทัดที่ 18 case เท่ากับ 3  
บรรทัดที่ 19 แสดงคำว่า " Good " ผ่านทางหน้าจอ  
บรรทัดที่ 20 หยุดการทำงาน  
บรรทัดที่ 21 case เท่ากับ 2  
บรรทัดที่ 22 แสดงคำว่า " Fail " ผ่านทางหน้าจอ  
บรรทัดที่ 23 หยุดการทำงาน  
บรรทัดที่ 24 case เท่ากับ 1  
บรรทัดที่ 25 แสดงคำว่า " Fail " ผ่านทางหน้าจอ  
บรรทัดที่ 26 หยุดการทำงาน  
บรรทัดที่ 27 case เท่ากับ 0  
บรรทัดที่ 28 แสดงคำว่า " Fail " ผ่านทางหน้าจอ  
บรรทัดที่ 29 หยุดการทำงาน  
บรรทัดที่ 30 case ที่นอกจากนี้  
บรรทัดที่ 31 แสดงคำว่า " Out of range " ผ่านทางหน้าจอ  
บรรทัดที่ 32 หยุดการทำงาน  
บรรทัดที่ 33 วงเล็บปีกกาปิด ของ switch-case  
บรรทัดที่ 34 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 35 วงเล็บปีกกาปิด

**ผลการรัน**

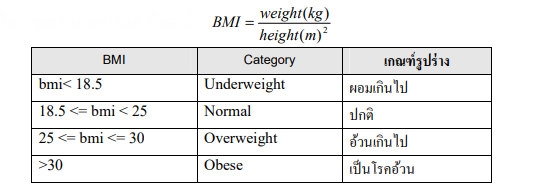


**สรุปผลการทดลอง**

จากโปรแกรมดังกล่าว เป็นโปรแกรมเพื่อหาระดับของคะแนนโดยการใช้การทำงานแบบ switch-case โดยเมื่อตรงกับค่าที่กำหนดของ case นั้น ๆ ก็จะทำการแสดงข้อความต่าง ๆ ผ่านทางหน้าจอ เช่น ระดับ 5 : Excellent, ระดับ 4 : Very Good, ระดับ 3 : Good, ระดับ 0-2 : Fail, นอกจากนั้น : Out of range

**Checkpoint 2**

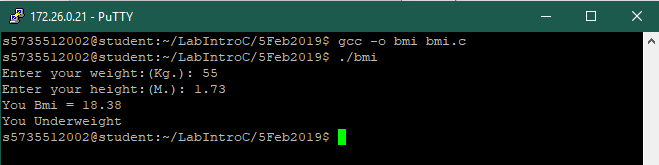
จงเขียนโปรแกรม bmi.c เพื่อรับค่าส่วนสูง(หน่วยเมตร) และน้ำหนัก(หน่วยกิโลกรัม) ของผู้ใช้แล้วคำนวณหาค่า BMI (Body Mass Index) ซึ่งจะเป็นตัวระบุว่าผู้ใช้มีรูปร่างอยู่ในเกณฑ์ใน สูตรคำนวณค่า BMI เป็นดังนี้



**Source code**

|  |  |
| --- | --- |
| 1. 2.  3.  4  5.  6.  7  8. 9. 10.  11.  12.  13.  14.  15.  16.  17.  18.  19.  20.  21. | //bmi.c  #include<stdio.h>  int main()  {  float weight,height,bmi=0;  printf("Enter your weight:(Kg.): ");  scanf("%f",&weight);  printf("Enter your height:(M.): ");  scanf("%f",&height);  bmi=weight/(height\*height);  printf("You Bmi = %.2f\n",bmi);  if(bmi<18.5)  printf("You Underweight\n");  else if(bmi>=18.5&&bmi<25)  printf("You Normal\n");  else if(bmi>=25&&bmi<=30)  printf("You Overweight\n");  else if(bmi>30)  printf("You Obese\n");  return 0;  } |

**อธิบาย Source Code**  
บรรทัดที่ 1 Comment  
บรรทัดที่ 2 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 3 ฟังก์ชันหลัก  
บรรทัดที่ 4 วงเล็บปีกกาเปิด  
บรรทัดที่ 5 ประกาศตัวแปรชนิด float ชื่อ weight, ชื่อ height และชื่อ bmi และกำหนดให้ 0  
บรรทัดที่ 6 แสดงคำว่า " Enter your weight:(Kg.): " ผ่านทางหน้าจอ  
บรรทัดที่ 7 ทำการรับค่าเป็น float เก็บไว้ที่ weight  
บรรทัดที่ 8 แสดงคำว่า " Enter your height:(M.): " ผ่านทางหน้าจอ  
บรรทัดที่ 9 ทำการรับค่าเป็น float เก็บไว้ที่ height  
บรรทัดที่ 10 bmi=weight/(height\*height);  
บรรทัดที่ 11 แสดงคำว่า " You Bmi = %.2f" ผ่านทางหน้าจอ  
บรรทัดที่ 12 เงื่อนไขแบบ if-else กรณี if เช็คเงื่อนไขว่า bmi น้อยกว่า 18.5  
บรรทัดที่ 13 แสดงคำว่า " You Underweight " ผ่านทางหน้าจอ  
บรรทัดที่ 14 กรณี else if เช็คเงื่อนไขว่า bmi มากกว่าหรือเท่ากับ 18.5 และ bmi น้อยกว่า 25  
บรรทัดที่ 15 แสดงคำว่า " You Normal " ผ่านทางหน้าจอ  
บรรทัดที่ 16 กรณี else if เช็คเงื่อนไขว่า bmi มากกว่าหรือเท่ากับ 25 และ bmi น้อยกว่าหรือเท่ากับ 30  
บรรทัดที่ 17 แสดงคำว่า " You Overweight " ผ่านทางหน้าจอ  
บรรทัดที่ 18 กรณี else if เช็คเงื่อนไขว่า bmi มากกว่า 30  
บรรทัดที่ 19 แสดงคำว่า " You Obese " ผ่านทางหน้าจอ  
บรรทัดที่ 20 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 21 วงเล็บปีกกาปิด  
  
**ผลการรัน**



**สรุปผลการทดลอง**

จากโปรแกรมดังกล่าว เป็นโปรแกรมหา BMI โดยมีการรับค่าน้ำหนักและส่วนสูง จากนั้นจะแสดงข้อความเพื่อว่าผู้ใช้มีรูปร่างอยู่ในเกณฑ์ใน เช่น You Underweight, You Normal, You Overweight หรือ You Obese

**งานท้ายการทดลอง**

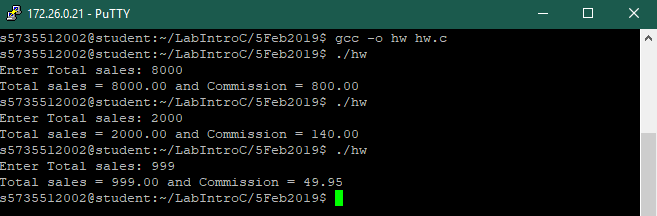
จงเขียนโปรแกรมเพื่อคำนวณค่านายหน้า (Sales commission) โดยโปรแกรมรับค่ายอดขายรวม (Total sales) ของพนักงานขายแล้วนำมาคำนวณหาค่านายหน้าโดยคิดเป็นเปอร์เซ็นต์จากยอดขาย แบ่งออกเป็น 3 ช่วง คือ หากยอดขายต่ำกว่า 1000 บาท ให้คิด 5% , 1000 บาท – 5000 บาท ให้คิด 7% และหากเกิน 5000 บาท ให้คิด 10% เช่น ยอดขาย 8000 บาท จะได้ค่านายหน้า 800 บาท, ยอดขาย 2000 บาท จะได้ค่านายหน้า 140 บาท เป็นต้น

**Source code**

|  |  |
| --- | --- |
| 1. 2.  3.  4  5.  6.  7  8. 9. 10.  11.  12.  13.  14.  15.  16. | //hw.c  #include<stdio.h>  int main()  {  float total,cal;  printf("Enter Total sales: ");  scanf("%f",&total);  if(total<1000)  cal = total\*0.05;  else if(total>=1000&&total<=5000)  cal =total\*0.07;  else if(total>5000)  cal = total\*0.10;  printf("Total sales = %.2f and Commission = %.2f \n",total,cal);  return 0;  } |

**อธิบาย Source Code**  
บรรทัดที่ 1 Comment  
บรรทัดที่ 2 เรียกใช้ library Standard Input/Output  
บรรทัดที่ 3 ฟังก์ชันหลัก  
บรรทัดที่ 4 วงเล็บปีกกาเปิด  
บรรทัดที่ 5 ประกาศตัวแปรชนิด float ชื่อ total แบะ ชื่อ cal  
บรรทัดที่ 6 แสดงคำว่า " Enter Total sales: " ผ่านทางหน้าจอ  
บรรทัดที่ 7 ทำการรับค่าเป็น float เก็บไว้ที่ total  
บรรทัดที่ 8 เงื่อนไขแบบ if-else กรณี if เช็คเงื่อนไขว่า total น้อยกว่า 1000  
บรรทัดที่ 9 คำนวณจากสูตร cal = total\*0.05;  
บรรทัดที่ 10 กรณี else if เช็คเงื่อนไขว่า total มากกว่า/เท่ากับ 1000 และ total น้อยกว่า/เท่ากับ 5000  
บรรทัดที่ 11 คำนวณจากสูตร cal =total\*0.07;  
บรรทัดที่ 12 กรณี else if เช็คเงื่อนไขว่า total มากกว่า 5000   
บรรทัดที่ 13 คำนวณจากสูตร cal = total\*0.10;  
บรรทัดที่ 14 แสดงคำว่า " Total sales = %.2f and Commission = %.2f " ผ่านทางหน้าจอ  
บรรทัดที่ 15 ส่งค่าเพื่อบอกว่าโปรแกรม Success แล้ว  
บรรทัดที่ 16 วงเล็บปีกกาปิด

**ผลการรัน**



**สรุปผลการทดลอง**

จากโปรแกรมดังกล่าว เป็นโปรแกรมคำนวณค่านายหน้า (Sales commission) โดยโปรแกรมรับค่ายอดขายรวม (Total sales) ของพนักงานขายแล้วนำมาคำนวณหาค่านายหน้าโดยคิดเป็นเปอร์เซ็นต์จากยอดขาย แบ่งออกเป็น 3 ช่วง คือ หากยอดขายต่ำกว่า 1000 บาท ให้คิด 5% , 1000 บาท – 5000 บาท ให้คิด 7% และหากเกิน 5000 บาท ให้คิด 10%