Let Us C

Manish

December 6, 2023

Chapter 1

Getting Started

[A] Which of the following are invalid variable names & why?

a BASICSALARY, _basic, mindovermatter, FLOAT, hELLO
are valid variable names.

b basic-hra, #MEAN, team'svictory, Plot#3
invalid variable names: contains special symbols

c 421, 2015_DDay
invalid due to: starting with numbers

d group., queue.
are invalid due to the period symbol.

e population in 2006, over time,
invalid due to: containing spaces

[B] Point out the errors, if any, in the following C statements:

a no variable initialized or int is an invalid variable name.

b no error

c no error

d cannot have multiple variables on the left side of the assignment operator.

e * is not used to multiply.

f spaces in the variable names.

g no error

h '**' is invalid operator

i invalid operator

j implicit multipication is not allowed (skipped * operator)

k cannot assign value to a constant.

l no error

m string must be enclosed in "(double quotes).

[C] Evaluate the following expressions and show their hierarchy.

a
g = 2/2 + 2*4/2 - 2 + 2.5/3
g = 1 + 8/2 - 2 + 0.83
g = 1 + 4 - 2 + 0.83
g = 3.83

b
on = 4*1/2 + 3/2*1 + 2 + 3.2
on = 4/2 + 1*1 + 2 + 3.2
on = 2 + 1 + 2 + 3.2
on = 8

c
s = 4*2/4 - 6/2 + 2/3*6/2
s = 8/4 - 3 + 0*6/2
s = 2 - 3 + 0/2
s = 2 - 3 + 0
s = -1

d
s = 1/3*4/4 - 6/2 + 2/3*6/3
s = 0*1 - 3 + 0*2
s = 0 - 3 + 0
s = -3

[D] Fill the following table for the expressions given below and then evaluate the result.

Operator	Left	Right	Remark
a) /	10	5 or 5/2/1	Left operand is unambiguous, Right is not
b) /	3	2	Left operand is unambiguous, Right is not
c) +	3	4	both are unambiguous

[E] Convert the following equations into correspondign C statements.

a Z = (8.8*(a+b)*2/c-0.5+2*a/(q+r)) / (a+b)*(1/m))

b X = (-b+(b*b)+24*a*c)/(2*a)

c R = (2*v + 6.22 * (c+d))/ (g+v)

d A = (7.7*b*(xy+a)/c - 0.8 + 2*b) / ((x+a)*(1/y))

[F] What would be the output of the following programs:

a 0 2 0.000000 2.000000

b a = 0 b = -6

c 1

d nn

nn
nn /n/n nn

e a = “value of a” b = “value of b entered by user”

f p = garbageValue q = garbageValue

[G] Pick up the correct alt for each of the following questions:

a Dennis Ritchie

b All the above

c a compiler

d ASCII form only

e 1 character

f ASCII value of Z

g All of the above

h 32767

i a number

j 3 + a = b

k / or *, - or +

l 6.6 / a

m ( )

n Each new C instruction has to be written on a separate line.

o 2

p 0

q 32768

r 6

s is used first

t at least one digit

u 1 character

v all the above

w keywords can be used as variable names

x constants on the right side of =

y */+-

z 0.2857

[H] Write C programs fro the following:

a
```
int main() {
  float basic, gross;
  scanf("%f", &basic);
  gross = basic*40/100 + basic*20/100;
  printf("gross sallary = %f\n", gross);
  return 0;
}
```

b
```
int main() {
  float dist;
  scanf("%f", &dist);
  printf("merters = %f\n", dist*1000);
  printf("feet = %f\n", dist*3.23);
  printf("inches = %f\n", dist*3.23*12);
  printf("centimeters = %f\n", dist*100000);
  return 0;
}
```

c
```
int main() {
  float s1, s2, s3, s4, s5, ttlMks;
  scanf("%f%f%f%f%f", &s1, &s3, &s3, &s4, &s5);
  ttlMks = s1+ s2+ s3+ s4+ s5;
  printf("aggegrate = %f\n", ttlMks);
  printf("percentage = %f\n", ttlMks*100/500);
  return 0;
}
```

d

```
int main() {
  float fDeg, cDeg;
  scanf("%f", &fDeg);
  cDeg = (fDeg-32) * 5.0/9;
  printf("centigrade = %f\n", cDeg);
  printf("percentage = %f\n", ttlMks*100/500);
  return 0;
}
```

e

```
int main() {
  float l, b, r;
  scanf("%f%f%f", &l, &b, &r);
  printf("rectangle perimeter = %f\n", l*b);
  printf("rectangle area = %f\n", 2*(l+b));
  printf("circle circumference = %f\n", 2*3.14*r);
  printf("circle area = %f\n", 3.14*r*r);
  return 0;
}
```

f

```
int main() {
  int c, d;
  scanf("%d%d", &c, &d);
  c += d;
  d = c - d;
  c = c - d;
  printf("swapped variables c = %d d = %d\n", c, d);
  return 0;
}
```

g

```
int main() {
  int num, sum = 0;
  scanf("%d", &num);
  sum += num%10;
  num /= 10;
  sum += num%10;
  num /= 10;
  sum += num%10;
  num /= 10;
  sum += num%10;
  num /= 10;
  sum += num;
  printf("sum = %d\n", sum);
  return 0;
}
```

h
```
int main() {
  int num, rev = 0;
  scanf("%d", &num);
  rev = rev*10 + num%10;
  num /= 10;
  rev = rev*10 + num%10;
  num /= 10;
  rev = rev*10 + num%10;
  num /= 10;
  rev = rev*10 + num%10;
  num /= 10;
  rev = rev*10 + num;
  printf("reversed number = %d\n", rev);
  return 0;
}
```

i
```
int main() {
  int num, sum = 0;
  scanf("%d", &num);
  sum = num%10 + num/1000;
  printf("sum of first & last digit = %d\n", sum);
  return 0;
}
```

j
```
int main() {
  int menPer = 52, literacy = 48, litMen = 35, pop = 80000;
  int illMen = (menPer-litMen) * pop / 100;
  int illWomen = ((100-53)-(literacy-litMen)) * pop / 100;
  printf("no of illiterate men = %d\n", illMen);
  printf("no of illiterate women = %d\n", illWomen);
  return 0;
}
```

k
```
int main() {
  int amt;
  scanf("%d", &amt);
  printf("no of 100's = %d\n", amt/100);
  amt %= 100;
  printf("no of 50's = %d\n", amt/50);
  amt %= 50;
  printf("no of 10's = %d\n", amt/10);
  return 0;
```

```
}
```

l

```
int main() {
  int price, profit;
  scanf("%d %d", &price, & profit);
  int cost = (price*15 - profit) / 15;
  printf("cost price per item = %d\n", cost);
  return 0;
}
```

m

```
int main() {
  int num;
  scanf("%d", &num);
  printf("num after adding 1 to each digit = %d\n", num+11111);
  return 0;
}
```

Chapter 2

The Decision Control Structure

if, if-else, Nested *if-elses*

[A] What would be the output of the following programs:

a garabageValue 200

b 300 200

c 10 20

d 3
5

e x and y are equal

f x = 10 y = 10 z = 0

g 0 50 0

h C is WOW

i a = 15 b = 15 c = 0

j 1 20 1

[B] Point out the errors, if any, in the following programs:

a semantic error: a = b in if condition instead of a == b.

b no errors: unnecessary curly brackets are used.

c no errors.

d syntax error: c doesn't support then keyword.

e syntax error: parantheses are necessary for if condition.

f syntax error: more than one operand on left of assigment operator

g syntax error: elseif in not a valid keyword.

h syntax error: c doesn't support then keyword.

i semantic error: address of variables must be given in scanf variable, otherwise garbage values will be used.

[C] Attempt the following:

a
```
int main() {
  int sp, cp;
  scanf("%d %d", &sp, &cp);
  if(cp < sp) {
    printf("profit made = %d\n", sp-cp);
  } else {
    printf("loss made = %d\n", cp-sp);
  }
  return 0;
}
```

b
```
int main() {
  int num;
  scanf("%d", &num);
  if(num % 2 == 0) {
    printf("even number.\n");
  } else {
    printf("odd number.\n");
  }
  return 0;
}
```

c
```
int main() {
  int year;
  scanf("%d", &year);
  if(year%400 == 0){
    printf("%d is a leap year!", year);
  } else {
    if(year%4 == 0) {
      if(year%100 == 0) {
```

```
        printf("%d is not a leap year!", year);
      } else {
        printf("%d is a leap year!", year);
      }
    } else {
      printf("%d is not a leap year!", year);
    }
  }
  return 0;
}
```

d

```
int main() {
  int day = 1, month = 13, year;
  scanf("%d", &year);
  year--;
  int k = year % 100;
  int j = year / 100;

  int dayOfWeek = (day + ((13 * (month + 1)) / 5)
                   + k + (k / 4) + (j / 4) - (2 * j)) % 7;

  if (dayOfWeek < 0) {
    dayOfWeek += 7;
  }

  if(dayOfWeek == 0)
    printf("Saturday \n");
  else if(dayOfWeek == 1)
    printf("Sunday \n");
  else if(dayOfWeek == 2)
    printf("Monday \n");
  else if(dayOfWeek == 3)
    printf("Tuesday \n");
  else if(dayOfWeek == 4)
    printf("Wednesday \n");
  else if(dayOfWeek == 5)
    printf("Thursday \n");
  else
    printf("Friday \n");
  return 0;
}
```

e

```
int main() {
  int num, rev = 0;
  scanf("%d", &num);
```

```
  int temp = num;
  rev = rev*10 + num%10;
  num /= 10;
  rev = rev*10 + num%10;
  num /= 10;
  rev = rev*10 + num%10;
  num /= 10;
  rev = rev*10 + num%10;
  num /= 10;
  rev = rev*10 + num;
  if (temp == rev) {
    printf("%d is equal to %d\n", num, rev);
  } else {
    printf("%d is not equal to %d\n", num, rev);
  }
  return 0;
}
```

f

```
int main() {
  int ram, shyam, ajay;
  scanf("%d%d%d", &ram, &shyam, &ajay);
  if (ram < shyam) {
    if (ram < ajay) {
      printf("ram is the youngest.");
    } else {
      printf("ajay is the youngest.");
    }
  } else {
    if (shyam < ajay) {
      printf("shyam is the youngest.");
    } else {
      printf("ajay is the youngest.");
    }
  }
  return 0;
}
```

g

```
int main() {
  int a, b, c;
  scanf("%d%d%d", &a, &b, &c);
  if (a+b+c == 180) {
    printf("is a valid triangle.");
  } else {
    printf("is not a valid triangle.");
  }
```

```
  return 0;
}
```

h
```
int main() {
  int num;
  scanf("%d", &num);
  if (num < 0) {
    printf("absolute value = %d\n", num*(-1));
  } else {
    printf("absolute value = %d\n", num);
  }
  return 0;
}
```

i
```
int main() {
  int l, b, area, perimeter;
  scanf("%d%d", &l, &b);
  area = l*b;
  perimeter = 2*(l+b);
  if (area < perimeter) {
    printf("area is less than perimeter\n");
  } else {
    printf("area is greater than perimeter\n");
  }
  return 0;
}
```

j
```
int main() {
  int x1, y1, x2, y2, x3, y3;
  scanf("%d%d%d%d%d%d", &x1, &y1, &x2, &y2, &x3, &y3);
  if ((x2-x2)*(y2-y1) == (x2-x1)*(y3-y2)) {
    printf("points lie on the same line.\n");
  } else {
    printf("points are not collinear.\n");
  }
  return 0;
}
```

k
```
int main() {
    float cx, cy, radius, px, py;
    printf("coordinates of the center of the circle (x, y): ");
    scanf("%f %f", &cx, &cy);
    printf("radius of the circle: ");
```

```
    scanf("%f", &radius);
    printf("Enter the coordinates of the point (x, y): ");
    scanf("%f %f", &px, &py);

    float distance = sqrt(pow(px - cx, 2) + pow(py - cy, 2));

    if (distance < radius) {
        printf("The point is inside the circle.\n");
    } else if (distance == radius) {
        printf("The point is on the circle.\n");
    } else {
        printf("The point is outside the circle.\n");
    }
    return 0;
}
```

l

```
int main() {
  int x, y;
  scanf("%d%d", &x, &y);
  if (x == 0) {
    if (y == 0) {
      printf("point lie on the origin.\n");
    }
    else {
      printf("point lie on the x axis.\n");
    }
  } else {
    if (x == 0) {
      printf("point lie on the origin.\n");
    }
    else {
      printf("point lie on the y axis.\n");
    }
  }
  return 0;
}
```

Logical operators

If a = 10, b = 12, c = 0, find the values of the expressions in the following table:

Expression	Value
a != 6 && b > 5	1

```
a == 9 || b < 3          0
! ( a < 10 )             1
! ( a > 5 && c )         1
5 && c != 8 || !c        1
```

[D] What would be the output of the following programs:

a Dean of the students affairs.

b Let us C

c w = 1 x = 0 y = 1 z = 1

d y = 1 z = 1

e Bennarivo

f 40

g Definitely C !

h 1 1

i z is big

j -1 1

k k = 0

[E] Point out the errors, if any, in the following programs:

a no errors

b varibales are not initialized with any values. garbage values will be used.

c syntax error: 'or' is used instead of '||'.

d errors:

in if conditions '=' is used instead of '==' (equality) operator

logical operator && misused as it requires boolean values on both side.

e syntax error: 'and' is used instead of '&&'

f errors:

variables are not initialized.

'&' is used instead of '&&'.

g errors:

if is closed using ';' causing the else to be orphaned.

h error: no errors

[**F**] Attempt the following: (pg: 89)

a
```
#include <stdio.h>
int main() {
  int year;
  scanf("%d", &year);
  if((year%400 == 0) || (year%4 == 0 && year%100 != 0)) {
    printf("%d is a leap year!", year);
  } else {
    printf("%d is not a leap year!", year);
  }
  return 0;
}
```

b
```
#include <stdio.h>

int main() {
  char ch;
  scanf("%c", &ch);
  if((ch > -1 && ch < 48) || (ch > 57 && ch < 65) ||
      (ch > 90 && ch < 97) || (ch > 122 && ch < 128)) {
    printf("%c is a special symbol!", ch);
  } else if (ch < 58) {
    printf("%c is a digit!", ch);
  } else if (ch < 91) {
    printf("%c is a capital letter!", ch);
  } else if (ch < 123) {
    printf("%c is a small letter!", ch);
  }
  return 0;
}
```

c
```
int main() {
  int ensured, age, city, premRate, maxAmt, health, male;

  printf("age: ");
  scanf("%d", &age);
  printf("are you male(0/1): ");
  scanf("%d", &male);
  printf("do you live in city(0/1): ");
  scanf("%d", &city);
  printf("are you in excellent health(0/1):");
  scanf("%d", &health);
  ensured = 1;
```

```
    if (health == 1 && (age < 36 && age > 24) && city == 1) {
      if (male == 1) {
        premRate = 4;
        maxAmt = 200000;
      } else {
        premRate = 3;
        maxAmt = 100000;
      }
    } else if (health == 0 && (age < 36 && age > 24)
                && city == 0 && male == 1) {
        premRate = 6;
        maxAmt = 10000;
    } else {
        ensured = 0;
    }
    if (ensured) {
      printf("You can be ensured for: \n
                premRate: %d && max amount: %d\n"
                , premRate, maxAmt);
    } else {
      printf("You cannot be ensured.\n");
    }
    return 0;
  }
```

d

```
int main() {
    int hardness, tstrength, grade, c1, c2, c3;
    float carbon;
    printf("hardness: ");
    scanf("%d", &hardness);
    printf("carbon content: ");
    scanf("%f", &carbon);
    printf("tensile strength: ");
    scanf("%d", &tstrength);

    c1 = (hardness > 50)? 1: 0;
    c2 = (carbon < 0.7)? 1: 0;
    c3 = (tstrength > 5600)? 1: 0;

    if (c1 && c2 && c3) {
      grade = 10;
    } else if (c1 && c2 && !c3) {
      grade = 9;
    } else if (!c1 && c2 && c3) {
      grade = 8;
```

```
    } else if (c1 && !c2 && c3) {
      grade = 7;
    } else if (c1 || c2 || c3) {
      grade = 6;
    } else {
      grade = 5;
    }

    printf("grade = %d\n", grade);

    return 0;
  }
```

e
```
int main() {
    int fine, days;

    printf("enter days: ");
    scanf("%d", &days);

    if (days < 5) {
      printf("fine  50 paise\n");
    } else if (days < 10) {
      printf("fine  1 rupee\n");
    } else if (days < 31) {
      printf("fine  10 rupees\n");
    } else {
      printf("your membership will be cancelled.");
    }

    return 0;
  }
```

f
```
i1nt main() {
    int s1, s2, s3;
    printf("enter the 3 sides of triangle:");
    scanf("%d %d %d", &s1, &s2, &s3);

    if (s1 + s2 > s3 || s1 + s3 > s2 || s3 + s2 > s1) {
      printf("is a valid triangle.\n");
    } else {
      printf("is not a valid triangle.\n");
    }

    return 0;
  }
```

g
```
int main() {
    int s1, s2, s3;
    printf("enter the 3 sides of triangle:");
    scanf("%d %d %d", &s1, &s2, &s3);

    if(s1 == s2 && s2 == s3) {
      printf("is an equilateral triangle.\n");
    } else if (s1 == s2 || s1 == s3 || s2 == s3) {
      printf("is an isosceles triangle.\n");
    } else if (s1*s1 == (s2*s2 + s3*s3) ||
               s2*s2 == (s1*s1 + s3*s3) ||
               s3*s3 == (s1*s1 + s2*s2)) {
      printf("is a right angled triangle.\n");
    } else {
      printf("is an scalene triangle.\n");
    }

    return 0;
  }
```

h
```
int main() {
    int time;
    printf("enter time take to complete the job:");
    scanf("%d", &time);

    if(time >= 2 && time < 3) {
      printf("worker is highly efficient.\n");
    } else if (time >= 3 && time < 4) {
      printf("worker is ordered to increase speed.\n");
    } else if (time >= 4 && time < 5) {
      printf("worker needs to be given training.\n");
    } else if (time > 5) {
      printf("worker is asked to leave company.\n");
    }

    return 0;
  }
```

i
```
int main() {
    int a, b;
    printf("enter percentage obtained in subject A & B:");
    scanf("%d%d", &a, &b);
```

```
    if((a >= 55 && b < 45) || (a < 55 && a >= 45 && b >=55)) {
      printf("passed\n");
    } else if (b < 45 && a >= 65) {
      printf("allowed to reapper in B.\n");
    } else {
      printf("Failed.\n");
    }
    return 0;
  }


j int main() {
    int order, stock, hasCredit;
    printf("enter values:\norder:");
    scanf("%d", &order);
    printf("stock available:");
    scanf("%d", &stock);
    printf("credit OK (0/1):");
    scanf("%d", &hasCredit);

    if(order <= stock && hasCredit == 1) {
      printf("supply has requirement.\n");
      } else if (hasCredit == 0 && order <= stock) {
      printf("send intimation.\n");
      } else if (hasCredit == 1 && order > stock) {
      printf("send avaliable stock && intimate about balance shipment.\n");
    }
    return 0;
  }
```

Conditional operators

[G] What would be the output of the following programs: (pg: 92)

a depends on the garbage value of num:

if num ¡ 0; then the output will be 0

otherwise; output will be the square of garbage value.

b 200

c Welcome

[H] Point out the errors, if any, in the following programs: (pg: 93)

a syntax error: in conditional statement ? is used instead of :.

b syntax error: 2 format specifiers are used but only one variable is passed in printf.

c no errors

d syntax error in terinary operator, : is not used.

e errors:

terinary orerator is not complete

: is used in front of pfintf func

invalid); is used

f no errors

g no errors

[I] Rewrite the following programs using conditional operators: (pg: 94)

a
```
main() {
  int x, min, max;
  scanf("\n%d %d", &max, &x);
  (x > max)? (max = x): (min = x);
}
```

b
```
main() {
  int code;
  scanf("%d", &code);
  (code > 1)? printf("\nJerusalem"):
              ((code < 1)? printf("\nEddie"): printf("\nC Brain"));
}
```

c
```
main() {
  float sal ;
  printf ("Enter the salary" ) ;
  scanf ( "%f", &sal ) ;
  (sal < 40000 && sal > 25000)? printf ( "Manager" ):
      ((sal < 25000 && sal > 15000)?
      printf ( "Accountant" ): printf ( "Clerk" ));
}
```

[J] Attempt the following: (pg: 95)

a
```
int main() {
  char ch;
  printf("enter character:");
  scanf("%c", &ch);
```

```
    ((ch > -1 && ch < 48) || (ch > 57 && ch < 65) ||
     (ch > 90 && ch < 97) || (ch > 122 && ch < 128))
      ? printf("special symbol\n")
      : (ch > 96 && ch < 123)
        ? printf("lower case alphabet\n")
        : printf("some other character\n");

    return 0;
  }
```

b
```
  int main() {
    int year;
    scanf("%d", &year);

    ((year%400 == 0) || (year%4 == 0 && year%100 != 0))
      ? printf("%d is a leap year!", year)
      : printf("%d is not a leap year!", year);

    return 0;
  }
```

c
```
  int main() {
    int a, b, c;
    scanf("%d%d%d", &a, &b, &c);

    (a > b && a > c)
      ? printf("%d is the biggest one.\n", a)
      : (b > a && b > c)
        ? printf("%d is the biggest one.\n", b)
        : printf("%d is the biggest one.\n", c);

    return 0;
  }
```

Chapter 3

The Loop Control Structure

while loop

[A] What would be the output of the following programs:

a j is not initialized with any value so it will use the garbageValue already present in it. Making the output uncertain.

b

```
1
2
3
4
5
6
7
8
9
10
```

c same as part a of this section.

d

```
0
```

e

```
0
```

f syntax error in while statement.

g

```
2 3 3
```

h

```
3 3 1
```

i

```
malyalam is a palindrome
malyalam is a palindrome
(... infinite loop)
```

j

```
A computer buff!
A computer buff!
(... infinite loop)
```

k

```
10
10
(... infinite loop)
```

l

```
1.100000
```

m

```
In while loop
In while loop
(... infinite loop)
```

n

```
Ascii value 0 Character
Ascii value 1 Character
...
...
Ascii value 127 Character
Ascii value -128 Character
Ascii value -127 Character
...
...
```

```
Ascii value -1 Character
Ascii value 0 Character
Ascii value 1 Character
...
(... infinite loop)
```

o

```
3 1
1 3
0 4
-1 5
```

p

```
4 0
3 1
```

[B] Attempt the following:

a

```
int main() {
  int hWorked = 0, overtime = 0;
  printf("enter number of hours worked by employees: ");
  scanf("%d", &hWorked);
  overtime = (hWorked > 40)? 12*(hWorked-40): 0;

  printf("overtime pay = %d\n", overtime);
  return 0;
}
```

b

```
int main() {
  int num = 0, fact = 1;
  printf("enter number: ");
  scanf("%d", &num);
  while(num > 0) {
    fact = fact*num;
    num--;
  }
  printf("factorial = %d\n", fact);
  return 0;
}
```

c

```
int main() {
  int num1 = 0, num2 = 0, res = 1;
  printf("enter 2 numbers: ");
```

```
  scanf("%d%d", &num1, &num2);
  while(num2 > 0) {
    res *= num1;
    num2--;
  }
  printf("num1 raised to the num2 = %d\n", res);
  return 0;
}
```

d

```
int main() {
  int x = 0;
  while(x <= 255) {
    printf("Ascii value %d = %c\n", x, x);
    x++;
  }
  return 0;
}
```

e

```
int main() {
  int num = 1, d1, d2, d3;
  while(num <= 500) {
    //the three digits of number 544 = d1d2d3
    d1 = num/100;
    d2 = num/10 % 10;
    d3 = num%10;
    if (d1*d1*d1 + d2*d2*d2 + d3*d3*d3 == num) {
      printf("%d ", num);
    }
    num++;
  }
  return 0;
}
```

f

```
int main() {
  int matchsticks = 21, user;
  int r = 1;
  while(matchsticks > 0) {
    printf("\nRound %d\n", r++);
    printf("your move: \t\t");
    scanf("%d", &user);
    matchsticks -= user;
    if(matchsticks <= 0) {
      printf("\n\nremaining matchsticks = 0\nYOU LOSE!!!\n\n");
      break;
```

```
    }
    printf("my move: \t\t%d", 5-user);
    matchsticks -= 5-user;
    printf("\nremaining matchsticks = %d\n", matchsticks);
  }
  return 0;
}
```

g

```
int main() {
  int num, pve, nve, zs, choice;
  pve = nve = zs = 0;
  choice = 1;
  while(choice == 1) {
    printf("\nyour number: ");
    scanf("%d", &num);
    if(num > 0) {
      pve++;
    } else if(num < 0) {
      nve++;
    } else {
      zs++;
    }
    printf("continue?(1/0)\t");
    scanf("%d", &choice);
  }
  printf("numbers entered: \n+ve = %d\n-ve = %d\n0 = %d\n", pve, nve, zs);
  return 0;
}
```

h

```
int main() {
  int num, oct = 0, digits = 1;
  printf("\nyour number: ");
  scanf("%d", &num);
  printf("octal equivalent of %d = ", num);
  while(num > 0) {
    oct = ((num%8) * digits) + oct;
    num /= 8;
    digits *= 10;
  }
  printf("%d\n", oct);
  return 0;
}
```

i

```
 int main() {
```

```
  int num, min, max, choice;
  printf("\nyour number: ");
  scanf("%d", &num);
  min = max = num;
  printf("\ncontinue? (1/0) ");
  scanf("%d", &choice);
  while(choice == 1) {
    printf("\nyour number: ");
    scanf("%d", &num);
    if(min > num) {
      min = num;
    } else if(max < num) {
      max = num;
    }
    printf("\ncontinue? (1/0) ");
    scanf("%d", &choice);
  }
  printf("range of entered numbers = %d\n", max-min);
  return 0;
}
```

for, break, continue, do-while

[C] What would be the output of the following programs:

a no output

b

```
2
3
4
5
6
```

c

```
2
5
```

d

```
A
A
A
A
A
```

[D] Answer the following:

a initialize loop counter
test
incrementing/decrementing counter

b arithmetic, relational, assignment

c a for loop

d at least once

e initialization, execution of body, testing

f 3 is not an infinite loop

g continue

[E] Attempt the following:

a
```
#include <stdio.h>
#include <math.h>

int main() {
  int num, i, sr, isPrime;
  for(num = 1; num <= 300; num++) {
    isPrime = 1;
    i = 2;
    sr = sqrt(num);
    for(i = 2; i <= sr; i++) {
      if(num%i == 0) {
        isPrime = 0;
        break;
      }
    }
    if(isPrime) {
      printf("%d ", num);
    }
  }
  return 0;
}
```

b
```
int main() {
  int i;
  for(i = 1; i <= 30000; i++) {
    printf("%c", 1);
    \\ printf("%s", "\u263a");
  }
  return 0;
}
```

c
```
int main() {
  int i, num;
  float sum = 0, fact;
  for(i = 1; i <= 7; i++) {
    num = i;
    fact = 1;
    while(num > 0) {
      fact = fact*num;
      num--;
    }
    sum += i/fact;
  }
  printf("sum of first 7 terms = %f\n", sum);
  return 0;
}
```

d
```
int main() {
  int i, j, k;
  for(i = 1; i <= 3; i++) {
    for(j = 1; j <= 3; j++) {
      for(k = 1; k <= 3; k++) {
        printf("%d%d%d\n", i, j, k);
      }
    }
  }
  return 0;
}
```

e
```
int main() {
  float i, x;
  int y;
  printf("\t\t\t1\t\t\t2\t\t\t3\t\t\t4\t\t\t5\t\t\t6");
  for(x = 5.5; x <= 12.5; x += 0.5) {
    printf("\n%4.1f\t", x);
    for(y = 1; y <= 6; y++) {
```

```
      i = 2 + (y + (0.5 * x));
      printf("%.2f\t", i);
    }
  }
  return 0;
}
```

f
```
int main() {
  int i, j;
  for(i = 0; i < 7; i++) {
    for(j = 0; j < 13; j++) {
      if((7-i <= j && j < 7) || (i-1 > j%7 && j/7)) {
        printf("  ");
        continue;
      }
      if(j < 7) {
        printf("%c ", 'A' + j);
      } else {
        printf("%c ", 'F' - j%7);
      }
    }
    printf("\n");
  }
  return 0;
}
```

g
```
int main() {
  int i;
  for(i = 0; i < 30000; i++) {
    //printf("%c%c", 3, 4);
    printf("%s%s", "\u2661", "\u2662");
  }
  return 0;
}
```

h
```
int main() {
  int i, num;
  scanf("%d", &num);
  for(i = 1; i <= 10; i++) {
    printf("%d * %2d = %3d\n", num, i, num*i);
  }
  return 0;
}
```

i
```
int main() {
  int i, j, num = 0;
  for(i = 0; i < 4; i++) {
    for(j = 0; j < 4; j++) {
      if(i+j < 3) {
        printf(" ");
      } else {
        printf("%d ", ++num);
      }
    }
    printf("\n");
  }
  return 0;
}
```

j
```
int main() {
  int i, j, n, nmr, r, nfact, nmrfact, rfact;
  for(i = 0; i < 5; i++) {
    for(j = 5-i; j > 0; j--) {
      printf(" ");
    }
    for(j = 0; j <= i; j++) {
        n = i; r = j; nmr = n-r;
        nfact = nmrfact = rfact = 1;
        while(n > 0) {
          nfact *= n;
          n--;
        }
        while(nmr > 0) {
          nmrfact *= nmr;
          nmr--;
        }
        while(r > 0) {
          rfact *= r;
          r--;
        }
        printf("%d ", nfact / (nmrfact*rfact));
    }
    printf("\n");
  }
  return 0;
}
```

k
```
int main() {
  int mCost = 6000, mEarning = 1000, mSalvage = 2000;
```

```
  int year = 0;
  float altEarn = 0, mEarn = 0, iRate = 12;
  while(altEarn >= mEarn) {
    altEarn += (altEarn + 4000) * iRate / 100;
    mEarn += 1000;
    year++;
  }
  printf("minimum life = %d\n", year);

  return 0;
}
```

l

```
int main() {
  float p, r, n, q, nq;
  double amount = 0, exp, expNq;
  int i = 0, j;
  while(i < 10) {
    scanf("%f%f%f%f", &p, &r, &n, &q);
    j = 0;
    exp = 1 + r/q;
    expNq = 1;
    nq = n*q;
    while(j < nq) {
      expNq *= exp;
      j++;
    }
    amount = p * expNq;
    printf("amount = %lf\n", amount);
    i++;
  }

  return 0;
}
```

m

```
int main() {
  float x;
  scanf("%f", &x);
  float exp, expn, nlog = (x-1)/x;
  expn =  exp = nlog;

  int i = 2;
  while(i < 8) {
    expn *= exp;
    nlog += expn/2;
    i++;
```

```
  }
  printf("natural log of first terms = %f\n", nlog);

  return 0;
}
```

Chapter 4

The Case Control Structure

[A] What would be the output of the following programs:

a

```
Heart
I thought one wears a suite
```

b

```
I am in case 3
```

c

```
Pure Simple Egghead!
```

d

```
Customers are dicey
Markets are pricey
Inverstors are moody
At least employees are good
```

e

```
Trapped
```

f

```
You entered a and b
```

g

```
Feeding fish
Weeking grass
mending roof
```

```
Just to survive
```

[B] Point out the errors, if any, in the following programs:

a
```
syntax error in case 0 & 1;
Also case statements are not allowed outside switch statement.
```

b
```
error: expression in case is not integer constant. (operand is not a constant)
```

c
```
error: quantity in switch is not an integer.
```

d
```
error: 2nd case statement is not an integer constant,
variables a and b are not considered constatns.
```

[C] Write a menu driven program which has following options:

1. Factorial of a number.
2. Prime or not
3. Odd or even
4. Exit

```
#include <stdio.h>
#include <math.h>

int main() {
  int choice, i, sr, num, fact;
  while(1) {
    printf("\n1. Factorial");
    printf("\n2. Prime");
    printf("\n3. Odd/Even");
    printf("\n4. Exit");
    printf("\nYour choice?");
    scanf("%d", &choice);

    switch (choice) {
      case 1:
        printf("\nenter number: ");
        scanf("%d", &num);
        fact = 1;
        while(num > 0) {
          fact = fact*num;
```

```
            num--;
          }
          printf("factorial = %d\n", fact);
          break;
        case 2:
          printf("\nenter number: ");
          scanf("%d", &num);
          i = 2;
          sr = sqrt(num);
          for(i = 2; i <= sr; i++) {
            if(num%i == 0) {
              printf("%d is a prime number.\n", num);
              break;
            }
          }
          break;
        case 3:
          printf("\nenter number: ");
          scanf("%d", &num);
          if(num%2 == 0) {
            printf("%d is an even number.\n", num);
          } else {
            printf("%d is an odd number.\n", num);
          }
          break;
        case 4:
          return 0;
      }
    }

    return 0;
  }
```

[D] Write a program to find the grace marks for a student using switch. The user should enter the class obtained by the student and the number of subjects he has failed in.:

```
#include <stdio.h>

int main() {
  int class, noOfSubs, grace = 0;
  printf("enter class obtained by student: ");
  scanf("%d", &class);
  printf("number of subjects failed in: ");
  scanf("%d", &noOfSubs);
```

```
  switch (class) {
    case 1:
      if(noOfSubs <= 3) {
        grace += noOfSubs * 5;
      }
      break;
    case 2:
      if(noOfSubs <= 2) {
        grace += noOfSubs * 4;
      }
      break;
    case 3:
      if(noOfSubs <= 1) {
        grace += noOfSubs * 5;
      }
      break;
  }

  printf("grace marks for student = %d\n", grace);

  return 0;
}
```