

Do NOT write on this test. Record all answers on the bubble sheet. **Closed book. No notes.** Work strictly from memory. **No calculators. No time limit. Scratch paper okay.**

On the following printf questions you are given a list of inputs. For each problem line determine which printf statement created the accompanying outputs. (␣ means space.)

Which of these printf statements created the outputs shown for each problem below? (x is int x;)

- (A) printf("␣␣␣␣␣%␣d␣",x); (D) printf("␣␣%+4d␣",x); (G) printf("␣%+04d␣",x);
 (B) printf("␣␣␣␣%0+3d␣",x); (E) printf("␣␣%-␣4d␣",x); (H) printf("␣%4d␣",x);
 (C) printf("␣␣%+3d␣",x); (F) printf("␣␣%-+5d␣",x); (I) printf("␣%5d␣",x);

inputs:	<u>2</u>	<u>-7</u>	<u>2120910699</u>	<u>-1798009096</u>
1/2p.	␣␣␣␣␣2␣	␣␣␣␣-7␣	␣2120910699␣	␣-1798009096␣
2/2p.	␣␣␣␣+2␣	␣␣␣␣-7␣	␣␣+2120910699␣	␣␣-1798009096␣
3/2p.	␣+002␣	␣-007␣	␣+2120910699␣	␣-1798009096␣
4/2p.	␣␣␣␣␣2␣	␣␣␣␣␣-7␣	␣␣␣␣␣2120910699␣	␣␣␣␣␣-1798009096␣

Which of these printf statements created the outputs shown for each problem below? (x is char * x;)

- (A) printf("␣␣␣␣%s␣",x); (D) printf("␣␣%2s␣",x); (G) printf("␣␣%s␣␣␣",x);
 (B) printf("␣␣␣␣%-3s␣",x); (E) printf("␣␣%3s␣",x); (H) printf("␣%-6s␣",x);
 (C) printf("␣␣%-2s␣",x); (F) printf("␣␣%4s␣",x); (I) printf("␣%6s␣",x);

inputs:	<u>""</u>	<u>"l"</u>	<u>"gz"</u>	<u>"vckw"</u>	<u>"zvbldc"</u>	<u>"wczwjfwp"</u>
5/2p.	␣␣␣␣␣	␣␣␣␣l␣	␣␣gz␣	␣␣vckw␣	␣␣zvbldc␣	␣␣wczwjfwp␣
6/2p.	␣␣␣␣␣	␣l␣␣␣␣␣	␣gz␣␣␣␣	␣vckw␣	␣zvbldc	␣wczwjfwp
7/2p.	␣␣␣␣␣	␣␣␣␣l␣	␣␣␣gz␣	␣␣␣vckw	␣␣␣zvbldc	␣␣␣wczwjfwp
8/2p.	␣␣␣␣␣	␣␣␣␣l␣	␣␣␣gz␣	␣␣vckw␣	␣␣zvbldc␣	␣␣wczwjfwp␣

Which of these printf statements created the outputs shown for each problem below? (x is double x;)

- (A) printf("␣␣%+12.6f",x); (D) printf("␣%0+12.0f␣",x); (G) printf("␣%0+12.6f␣",x);
 (B) printf("␣␣%10.0f␣",x); (E) printf("␣%12f␣",x); (H) printf("␣%014.2f",x);
 (C) printf("␣␣%11.4f␣",x); (F) printf("␣%+14.2f",x); (I) printf("␣%13.6f␣",x);

inputs:	<u>9</u>	<u>2.95</u>	<u>3.6074</u>	<u>-75048.950698</u>
9/2p.	+0009.000000␣	+0002.950000␣	+0003.607400␣	-75048.950698␣
10/2p.	␣␣␣␣␣9.000000␣	␣␣␣␣␣2.950000␣	␣␣␣␣␣3.607400␣	␣-75048.950698␣
11/2p.	00000000009.00	00000000002.95	00000000003.61	-0000075048.95
12/2p.	␣␣␣␣␣9.000000␣	␣␣␣␣␣2.950000␣	␣␣␣␣␣3.607400␣	-75048.950698␣

Precedence: What is the value of each expression? Mark (I) for error, (J) for none of the above.

- 13/1p. $4-5\%3*2/4$ (A) -15 (B) -2 (C) 0 (D) 2 (E) 3 (F) 4 (G) 34 (H) 98
 14/1p. $4-8!=1==0*2$ (A) -80 (B) -7 (C) 2 (D) 3 (E) 4 (F) 6 (G) 8 (H) 23
 15/1p. $5-5\&\&6||5+1$ (A) -99 (B) -71 (C) -68 (D) 1 (E) 2 (F) 3 (G) 4 (H) 30
 16/1p. $4*2+1-2*9$ (A) -60 (B) -54 (C) -31 (D) -28 (E) -1 (F) 36 (G) 63 (H) 90
 17/1p. $6/5/5*7-9$ (A) -12 (B) -9 (C) -3 (D) 0 (E) 2 (F) 33 (G) 50 (H) 61
 18/1p. $0-5\%2+7+7$ (A) -15 (B) -5 (C) -1 (D) 0 (E) 11 (F) 13 (G) 15 (H) 59
 19/1p. $6+9/0-5+3$ (A) -64 (B) -8 (C) -2 (D) 1 (E) 4 (F) 6 (G) 7 (H) 23
 20/1p. $4/7-9-7-3$ (A) -12 (B) -5 (C) -4 (D) -3 (E) -1 (F) 0 (G) 1 (H) 2
 21/1p. $3/6!=9\&\&5+8$ (A) -66 (B) -40 (C) 1 (D) 3 (E) 9 (F) 11 (G) 26 (H) 63
 22/1p. $8/9-6\%8*2$ (A) -12 (B) -6 (C) -3 (D) 4 (E) 10 (F) 37 (G) 65 (H) 75
 23/1p. $7/8-3+4*7$ (A) -49 (B) -31 (C) -1 (D) 1 (E) 7 (F) 25 (G) 29 (H) 35
 24/1p. $4*1+6*3/2$ (A) 10 (B) 11 (C) 13 (D) 15 (E) 28 (F) 36 (G) 42 (H) 94
 25/1p. $4/6\%5-8+2$ (A) -95 (B) -19 (C) -8 (D) -6 (E) -1 (F) 0 (G) 1 (H) 2
 26/1p. $8-5>=6>7+1$ (A) -65 (B) -20 (C) -6 (D) 0 (E) 1 (F) 6 (G) 7 (H) 8
 27/1p. $4+7+9\%7\%4$ (A) -25 (B) -7 (C) 1 (D) 2 (E) 5 (F) 6 (G) 11 (H) 48
 28/1p. $2+6\%8-2*7$ (A) -93 (B) -14 (C) -6 (D) -4 (E) 2 (F) 8 (G) 30 (H) 42
 29/1p. $3/9\&\&7!=7-2$ (A) -3 (B) -1 (C) 1 (D) 3 (E) 6 (F) 58 (G) 65 (H) 75
 30/1p. $8-7\%3+1-7$ (A) -6 (B) -5 (C) -2 (D) -1 (E) 1 (F) 10 (G) 13 (H) 77
 31/1p. $6*0||6>2-3$ (A) -53 (B) -40 (C) -18 (D) -12 (E) 1 (F) 3 (G) 6 (H) 44

How many times does the body of the loop execute? (Mark 9 if 9 or more.)

- 32/2p. `int t=-1; while(--t < 5) body;`
 33/2p. `int g=1; do body; while(g-- <= 7);`
 34/2p. `int g=6; do body; while(g++ != 11);`
 35/2p. `int u=6; do body; while(u++ < 10);`
 36/2p. `int g; for(g=0; g!=5; g++) body;`
 37/2p. `int p; for(p=-9; p>-10; --p) body;`
 38/2p. `int z=-8; while(++z < -2) body;`
 39/2p. `int t=0; while(++t < 3) body;`

40/3p. Give a tight big-oh $\Theta()$ bound on the running time $T(n)$ of this program.

Assume `atoi`, `simpleStatement`, and `simpleCompare` each run in $\Theta(1)$ time.

- (A) $n^2\sqrt{n}$ (C) $n\sqrt{n}\lg n$ (E) n (G) \sqrt{n} (I) $\lg n$
 (B) n^2 (D) $n\lg n$ (F) $\sqrt{n}\lg n$ (H) $\lg^2 n$ (J) 1

```
int main ( int argc, char * * argv ) {
    int n = atoi(argv[1]);
    if ( simpleCompare ) {
        i = 1; do {
            simpleStatement;
            i++; } while ( i * i < n );
    } else {
        simpleStatement;
    }
    return 0; }
```

41/3p. Give a tight big-oh $\Theta()$ bound on the running time $T(n)$ of this program.

Assume `atoi`, `simpleStatement`, and `simpleCompare` each run in $\Theta(1)$ time.

- (A) $n^2\sqrt{n}$ (C) n^2 (E) $n\sqrt{n}$ (G) n (I) \sqrt{n}
 (B) $n^2\lg n$ (D) $n\sqrt{n}\lg n$ (F) $n\lg n$ (H) $\sqrt{n}\lg n$ (J) 1

```
int main ( int argc, char * * argv ) {
    int n = atoi(argv[1]);
    if ( simpleCompare ) {
        g = 1; do {
            f = n; while ( f > 1 ) {
                simpleStatement;
                f--; }
            g += 2; } while ( g * g < n );
    } else {
        simpleStatement;
    }
    return 0; }
```

42/3p. Give a tight big-oh $\Theta()$ bound on the running time $T(n)$ of this program.

Assume `atoi`, `simpleStatement`, and `simpleCompare` each run in $\Theta(1)$ time.

- (A) $n^2\lg n$ (C) $n\sqrt{n}$ (E) n (G) \sqrt{n} (I) $\lg n$
 (B) n^2 (D) $n\lg n$ (F) $\sqrt{n}\lg n$ (H) $\lg^2 n$ (J) 1

```
int main ( int argc, char * * argv ) {
    int n = atoi(argv[1]);
    f = n; do {
        if ( simpleCompare ) {
            g = 1; while ( g * g < n ) {
                simpleStatement;
                g++; }
        } else {
            simpleStatement;
        }
        f /= 5; } while ( f > 1 );
    return 0; }
```

43/5p. Give a tight big-oh $\Theta()$ bound on the running time $T(n)$ of this program.

Assume `atoi`, `simpleStatement`, and `simpleCompare` each run in $\Theta(1)$ time.

- (A) $n^2\sqrt{n}$ (C) n^2 (E) $n\sqrt{n}$ (G) n (I) \sqrt{n}
 (B) $n^2\lg n$ (D) $n\sqrt{n}\lg n$ (F) $n\lg n$ (H) $\sqrt{n}\lg n$ (J) $\lg^3 n$

```
int main ( int argc, char * * argv ) {
  int n = atoi(argv[1]);
  e = 1; while ( e < n ) {
    if ( simpleCompare ) {
      k = 1; do {
        f = 1; while ( f < n ) {
          if ( simpleCompare ) {
            simpleStatement;
          }
          f++; }
        k *= 2; } while ( k < n );
    } else {
      if ( simpleCompare ) {
        if ( simpleCompare ) {
          simpleStatement;
        }
      }
    }
    e += 2; }
  return 0; }
```

44/5p. Give a tight big-oh $\Theta()$ bound on the running time $T(n)$ of this program.

Assume `atoi`, `simpleStatement`, and `simpleCompare` each run in $\Theta(1)$ time.

- (A) n^3 (C) n^2 (E) $n\sqrt{n}$ (G) n (I) \sqrt{n}
 (B) $n^2\lg n$ (D) $n\sqrt{n}\lg n$ (F) $n\lg n$ (H) $\sqrt{n}\lg n$ (J) 1

```
int main ( int argc, char * * argv ) {
  int n = atoi(argv[1]);
  for ( c = 1 ; c * c < n ; c += 3 ) {
    for ( e = n ; e > 1 ; e -= 10 ) {
      if ( simpleCompare ) {
        if ( simpleCompare ) {
          for ( i = 1 ; i < n ; i *= 5 ) {
            simpleStatement;
          }
        } else {
          simpleStatement;
        }
      } else {
        simpleStatement;
      }
    }
  }
  return 0; }
```

45/5p. Give a tight big-oh $\Theta()$ bound on the running time $T(n)$ of this program.

Assume `atoi`, `simpleStatement`, and `simpleCompare` each run in $\Theta(1)$ time.

- (A) n^7 (C) $n^4\sqrt{n}\lg^2 n$ (E) $n^3\lg^3 n$ (G) $n\sqrt{n}\lg n$ (I) $n\lg n$
 (B) $n^5\lg^2 n$ (D) $n^3\sqrt{n}$ (F) $n^3\lg^2 n$ (H) $n\lg^5 n$ (J) \sqrt{n}

```
int main ( int argc, char * * argv ) {
  int n = atoi(argv[1]);
  for ( b = 1 ; b < n ; b *= 2 ) {
    h = 1; while ( h * h < n ) {
      if ( simpleCompare ) {
        f = 1; do {
          for ( e = n ; e > 1 ; e -= 10 ) {
            if ( simpleCompare ) {
              a = n; do {
                for ( d = n ; d > 1 ; d /= 2 ) {
                  c = n; do {
                    simpleStatement;
                    c -= 1; } while ( c > 1 );
                  }
                a /= 5; } while ( a > 1 );
            } else {
              i = 1; while ( i * i < n ) {
                simpleStatement;
                i++; }
            }
          }
          f += 1; } while ( f * f < n );
        } else {
          if ( simpleCompare ) {
            for ( k = 1 ; k < n ; k += 1 ) {
              if ( simpleCompare ) {
                if ( simpleCompare ) {
                  simpleStatement;
                } else {
                  simpleStatement;
                }
              } else {
                simpleStatement;
              }
            }
          } else {
            g = n; while ( g > 1 ) {
              j = n; while ( j > 1 ) {
                simpleStatement;
                j -= 2; }
              g /= 2; }
          }
        }
      h++; }
    }
  return 0; }
```

Matching: Which Perl regular expression commands have what meaning? (If no match mark J.)

(A) \$ (B) \odd (C) \B (D) \b (E) \e (F) \n (G) \xdd (H) ^ (I) esc

- 46/1p. the escape character
- 47/1p. word boundary
- 48/1p. octal dd
- 49/1p. end of string
- 50/1p. newline

True or False: does the string match the regular expression?

- 51/1p. Does the string "kaa" match the regular expression "ka?|bu"?
- 52/1p. Does the string "qcq" match the regular expression "qt?cq"?
- 53/1p. Does the empty string match the regular expression "up+"?
- 54/1p. Does the string "rnqwrnqw" match the regular expression "(rn*qw)+"?
- 55/1p. Does the string "yg" match the regular expression "(yb?gu)+"?
- 56/1p. Does the string "gy" match the regular expression "(gy)?"?
- 57/1p. Does the string "byzyzz" match the regular expression "b|(yz*|uu)?"?
- 58/1p. Does the string "wnfkkppfkkpp" match the regular expression "wn+|(fk*|pp)+"?
- 59/1p. Does the string "bq" match the regular expression "((tc)+|bq|k*gt)*"?
- 60/1p. Does the string "r" match the regular expression "r*|sa((yy)?bu)+"?
- 61/1p. Does the string "tttt" match the regular expression "ay+|g|(c*|t+)?"?
- 62/1p. Does the string "aghyf" match the regular expression "au*gh(yf)+"?

Total points 100.

Answer Key (points per line)

1 (2).	I	32 (2).	9
2 (2).	D	33 (2).	9
3 (2).	G	34 (2).	6
4 (2).	A	35 (2).	5
5 (2).	D	36 (2).	5
6 (2).	H	37 (2).	1
7 (2).	B	38 (2).	5
8 (2).	E	39 (2).	2
9 (2).	G	40 (3).	$G(\sqrt{n})$
10 (2).	E	41 (3).	$E(n\sqrt{n})$
11 (2).	H	42 (3).	$F(\sqrt{n} \lg n)$
12 (2).	I	43 (5).	$B(n^2 \lg n)$
13 (1).	D (2)	44 (5).	$D(n\sqrt{n} \lg n)$
14 (1).	J (0)	45 (5).	$E(n^3 \lg^3 n)$
15 (1).	D (1)	46 (1).	E
16 (1).	J (-9)	47 (1).	D
17 (1).	B (-9)	48 (1).	B
18 (1).	F (13)	49 (1).	A
19 (1).	I (error)	50 (1).	F
20 (1).	J (-19)	51 (1).	false
21 (1).	C (1)	52 (1).	true
22 (1).	A (-12)	53 (1).	false
23 (1).	F (25)	54 (1).	true
24 (1).	C (13)	55 (1).	false
25 (1).	D (-6)	56 (1).	true
26 (1).	D (0)	57 (1).	false
27 (1).	J (13)	58 (1).	false
28 (1).	C (-6)	59 (1).	true
29 (1).	J (0)	60 (1).	true
30 (1).	E (1)	61 (1).	true
31 (1).	E (1)	62 (1).	true

Total points 100.