**1. struct student**

**{**

**static int a;**

**register int b;**

**auto int c;**

**extern int d;**

**}s={6,7,8,9};**

**void main()**

**{**

**printf("%d %d % %d",s.a,s.b,s.c,s.d);**

**}**

**2. struct student**

**{**

**int roll;**

**int cgpa;**

**int sgpa[8];**

**};**

**void main()**

**{**

**struct student s={12,8,7,2,5,9};**

**int \*ptr;**

**ptr=(int \*)&s;**

**printf("%d",\*(ptr+3));**

**}**

**3. struct game**

**{**

**int level;**

**int score;**

**struct player**

**{**

**char \*name;**

**}g2={"anil"};**

**}g3={10,200};**

**void main()**

**{**

**struct game g1=g3;**

**printf("%d %d %s",g1.level,g1.score,g1.g2.name);**

**}**

**4. struct game**

**{**

**int level;**

**int score;**

**struct player**

**{**

**char \*name;**

**}g2;**

**}g1;**

**void main()**

**{**

**printf("%d %d %s",g1.level,g1.score,g1.g2.name);**

**}**

**5. static struct student**

**{**

**int a;**

**int b;**

**int c;**

**int d;**

**}s1={6,7,8,9},s2={4,3,2,1},s3;**

**void main()**

**{**

**s3=s1+s2;**

**printf("%d %d %d %d",s3.a,s3.b,s3.c,s3.d);**

**}**

**6. extern struct student**

**{**

**int a;**

**int b;**

**int c;**

**int d;**

**}s={6,7,8,9};**

**void main()**

**{**

**printf("%d %d %d %d",s.a,s.b,s.c,s.d);**

**}**

**7. struct marks**

**{**

**int p:3;**

**int c:3;**

**int m:2;**

**};**

**void main(){**

**struct marks s={2,-6,5};**

**printf("%d %d %d",s.p,s.c,s.m);**

**}**

**8. enum actor{**

**SeanPenn=5,**

**AlPacino=-2,**

**GaryOldman,**

**EdNorton**

**};**

**void main(){**

**enum actor a=0;**

**switch(a){**

**case SeanPenn:**

**printf("Kevin Spacey");**

**break;**

**case AlPacino:**

**printf("Paul Giamatti");**

**break;**

**case GaryOldman:**

**printf("Donald Shuterland");**

**break;**

**case EdNorton:**

**printf("Johnny Depp");**

**}**

**}**

**}**

**9. void main()**

**{**

**struct employee**

**{**

**unsigned id: 8;**

**unsigned sex:1;**

**unsigned age:7;**

**};**

**struct employee emp1={203,1,23};**

**printf("%d\t%d\t%d",emp1.id,emp1.sex,emp1.age);**

**}**

**10.void main()**

**{**

**struct bitfield**

**{**

**unsigned a:5;**

**unsigned c:5;**

**unsigned b:6;**

**}bit;**

**char \*p;**

**struct bitfield \*ptr,bit1={1,3,3};**

**p=&bit1;**

**p++;**

**printf("%d",\*p);**

**}**

**11.void main()**

**{**

**struct bitfield**

**{**

**signed int a:3;**

**unsigned int b:13;**

**unsigned int c:1;**

**};**

**struct bitfield bit1={2,14,1};**

**printf("%d",sizeof(bit1));**

**}**

**12.void main()**

**{**

**struct field**

**{**

**int a;**

**char b;**

**}bit;**

**struct field bit1={5,'A'};**

**char \*p=&bit1;**

**\*p=45;**

**printf("\n%d",bit1.a);**

**}**

**13.void main()**

**{**

**struct world**

**{**

**int a;**

**char b;**

**struct india**

**{**

**char c;**

**float d;**

**}p;**

**};**

**struct world st ={1,'A','i',1.8};**

**printf("%d %c %c %f",st.a,st.b,st.p.c,st.p.d);**

**}**

**14.void main()**

**{**

**struct india**

**{**

**char c;**

**float d;**

**};**

**struct world**

**{**

**int a[3];**

**char b;**

**struct india orissa;**

**};**

**struct world st ={{1,2,3},'P','q',1.4};**

**printf("%d %c %c %f",st.a[1],st.b,st.orissa.c,st.orissa.d);**

**}**

**15.Which of the following statements is incorrect**

**(a) typedef struct new**

**{**

**int n1;**

**char n2;**

**} DATA;**

**(b) typedef struct**

**{**

**int n3;**

**char \*n4;**

**}ICE;**

**(c) typedef union**

**{**

**int n5;**

**float n6;**

**} UDT;**

**(d) #typedef union**

**{**

**int n7;**

**float n8;**

**} TUDAT;**

**16.struct numnam**

**{**

**int no;**

**char name[25];**

**}**

**struct numnam n1[]={{12,"Fred"},{15,"Martin"},{8,"Peter"},{11,Nicholas"}};**

**...........**

**...........**

**printf("%d%d",n1[2].no,((\*(n1 + 2).no)) + 1);**

**what does the above statement print?**

**(a) 8,9 (b) 9,9 (c) 8,8 (d) 8,unpredictable value**

**17.**

**const enum Alpha{**

**X,**

**Y=5,**

**Z**

**}p=10;**

**int main(){**

**enum Alpha a,b;**

**a= X;**

**b= Z;**

**printf("%d",a+b-p);**

**}**

**18.enum A{**

**x,y=5,**

**enum B{**

**p=10,q**

**}varp;**

**}varx;**

**void main(){**

**printf("%d %d",x,varp.q);**

**}**

**19.Which of the following is not derived data type in c?**

**(A) Function**

**(B) Pointer**

**(C) Enumeration**

**(D) Array**

**(E) All are derived data type**

**20.extern enum cricket x;**

**void main(){**

**printf("%d",x);**

**}**

**21.const enum cricket{**

**Taylor,**

**Kallis=17,**

**Chanderpaul**

**}x=Taylor|Kallis&Chanderpaul;**

**What is the value of x.**

**(A) 0**

**(B) 15**

**(C) 16**

**(D) 17**

**(E) Compilation error**

**22.const enum Alpha{**

**X,**

**Y=5,**

**Z**

**}p=10;**

**int main(){**

**enum Alpha a,b;**

**a= X;**

**b= Z;**

**printf("%d",a+b-p);**

**}**

**23.struct ABC{**

**int a;**

**float b;**

**char c;**

**};**

**int main(){**

**struct ABC \*ptr=(struct ABC \*)0;**

**ptr++;**

**printf("%d",\*ptr);**

**}**

**24.Which of the following is an incorrect syntax to pass by reference**

**a member of a structure in a function?**

**(Assume: struct temp{int a;}s;)**

**a) func(&s.a);**

**b) func(&(s).a);**

**c) func(&(s.a));**

**d) None of the mentioned**

**25.Which of the following structure declaration doesn’t require pass-**

**by-reference?**

**a) struct{int a;}s;**

**main(){}**

**b) struct temp{int a;};**

**main(){**

**struct temp s;**

**}**

**c) struct temp{int a;};**

**main(){}**

**struct temp s;**

**d) None of the mentioned**

**26.For the following function call which option is not possible?**

**func(&s.a); //where s is a variable of type struct and a is the**

**member of the struct.**

**a) Compiler can access entire structure from the function.**

**b) Individual member’s address can be displayed in structure.**

**c) Individual member can be passed by reference in a function.**

**d) Both ( b) and (c).**

**27.struct temp**

**{**

**int a;**

**} s;**

**void change(struct temp);**

**main()**

**{**

**s.a = 10;**

**change(s);**

**printf("%d\n", s.a);**

**}**

**void change(struct temp s)**

**{**

**s.a = 1;**

**}**

**28.struct p**

**{**

**int x;**

**int y;**

**};**

**int main()**

**{**

**struct p p1[] = {1, 92, 3, 94, 5, 96};**

**struct p \*ptr1 = p1;**

**int x = (sizeof(p1) / 5);**

**if (x == 3)**

**printf("%d %d\n", ptr1->x, (ptr1 + x - 1)->x);**

**else**

**printf("false\n");**

**}**

**a) Compile time error**

**b) 1 5**

**c) Undefined behaviour**

**d) false**

**29.struct student**

**{**

**char \*c;**

**};**

**void main()**

**{**

**struct student m;**

**struct student \*s = &m;**

**s->c = "hello";**

**printf("%s", s->c);**

**}**

**a) hello**

**b) Run time error**

**c) Nothing**

**d) Depends on compiler**

**30. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*c;**

**};**

**void main()**

**{**

**struct student \*s;**

**s->c = "hello";**

**printf("%s", s->c);**

**}**

**a) hello**

**b) Segmentation fault**

**c) Run time error**

**d) Nothing**

**31. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*c;**

**};**

**void main()**

**{**

**struct student m;**

**struct student \*s = &m;**

**s->c = "hello";**

**printf("%s", m.c);**

**}**

**a) Run time error**

**b) Nothing**

**c) hello**

**d) Varies**

**32. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*c;**

**};**

**void main()**

**{**

**struct student m;**

**struct student \*s = &m;**

**(\*s).c = "hello";**

**printf("%s", m.c);**

**}**

**a) Run time error**

**b) Nothing**

**c) Varies**

**d) hello**

**33. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*c;**

**};**

**void main()**

**{**

**struct student n;**

**struct student \*s = &n;**

**(\*s).c = "hello";**

**printf("%p\n%p\n", s, &n);**

**}**

**a) Different address**

**b) Run time error**

**c) Nothing**

**d) Same address**

**34. What is the output of this C code?**

**#include <stdio.h>**

**struct p**

**{**

**int x[2];**

**};**

**struct q**

**{**

**int \*x;**

**};**

**int main()**

**{**

**struct p p1 = {1, 2};**

**struct q \*ptr1;**

**ptr1->x = (struct q\*)&p1.x;**

**printf("%d\n", ptr1->x[1]);**

**}**

**a) Compile time error**

**b) Segmentation fault/code crash**

**c) 2**

**d) 1**

**35. What is the output of this C code?**

**#include <stdio.h>**

**struct p**

**{**

**int x[2];**

**};**

**struct q**

**{**

**int \*x;**

**};**

**int main()**

**{**

**struct p p1 = {1, 2};**

**struct q \*ptr1 = (struct q\*)&p1;**

**ptr1->x = (struct q\*)&p1.x;**

**printf("%d\n", ptr1->x[0]);**

**}**

**a) Compile time error**

**b) Undefined behaviour**

**c) Segmentation fault/code crash**

**d) 1**

**36. What is the output of this C code?**

**#include <stdio.h>**

**struct p**

**{**

**int x;**

**int y;**

**};**

**int main()**

**{**

**struct p p1[] = {1, 2, 3, 4, 5, 6};**

**struct p \*ptr1 = p1;**

**printf("%d %d\n", ptr1->x, (ptr1 + 2)->x);**

**}**

**a) 1 5**

**b) 1 3**

**c) Compile time error**

**d) 1 4**

**37. What is the output of this C code?**

**#include <stdio.h>**

**struct p**

**{**

**int x;**

**char y;**

**};**

**int main(){**

**struct p p1[] = {1, 92, 3, 94, 5, 96};**

**struct p \*ptr1 = p1;**

**int x = (sizeof(p1) / sizeof(struct p));**

**printf("%d %d\n", ptr1->x, (ptr1 + x - 1)->x);**

**}**

**a) Compile time error**

**b) Undefined behaviour**

**c) 1 3**

**d) 1 5**

**38. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*c;**

**struct student \*point;**

**};**

**void main()**

**{**

**struct student s;**

**struct student m;**

**s.c = m.c = "hi";**

**m.point = &s;**

**(m.point)->c = "hey";**

**printf("%s\t%s\t", s.c, m.c);**

**}**

**a) hey hi**

**b) hi hey**

**c) Run time error**

**d) hey hey**

**39. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*c;**

**struct student \*point;**

**};**

**void main()**

**{**

**struct student s;**

**struct student m;**

**m.point = s;**

**(m.point)->c = "hey";**

**printf("%s", s.c);**

**}**

**a) Nothing**

**b) Compile time error**

**c) hey**

**d) Varies**

**40. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*c;**

**struct student point;**

**};**

**void main()**

**{**

**struct student s;**

**s.c = "hello";**

**printf("%s", s.c);**

**}**

**a) hello**

**b) Nothing**

**c) Varies**

**d) Compile time error**

**41. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*c;**

**struct student \*point;**

**};**

**void main()**

**{**

**struct student s;**

**printf("%d", sizeof(s));**

**}**

**a) 5**

**b) 9**

**c) 8**

**d) 16**

**42. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*c;**

**struct student \*point;**

**};**

**void main()**

**{**

**struct student s;**

**struct student \*m = &s;**

**printf("%d", sizeof(student));**

**}**

**a) Compile time error**

**b) 8**

**c) 5**

**d) 16**

**43. What is the output of this C code?**

**#include <stdio.h>**

**struct p**

**{**

**int x;**

**char y;**

**struct p \*ptr;**

**};**

**int main()**

**{**

**struct p p = {1, 2, &p};**

**printf("%d\n", p.ptr->x);**

**return 0;**

**}**

**a) Compile time error**

**b) Undefined behaviour**

**c) 1**

**d) 2**

**44. What is the output of this C code?**

**#include <stdio.h>**

**typedef struct p \*q;**

**struct p**

**{**

**int x;**

**char y;**

**q ptr;**

**};**

**typedef struct p \*q;**

**int main()**

**{**

**struct p p = {1, 2, &p};**

**printf("%d\n", p.ptr->x);**

**return 0;**

**}**

**a) Compile time error**

**b) 1**

**c) Undefined behaviour**

**d) Address of p**

**45. Presence of loop in a linked list can be tested by the compiler**

**by.**

**a) Traveling the list, if NULL is encountered no loop exists**

**b) Comparing the address of nodes by address of every other node**

**c) Comparing the the value stored in a node by a value in every**

**other node**

**d) Both ( a) and (b).**

**46. What is the output of this C code?**

**#include <stdio.h>**

**typedef struct p \*q;**

**int main()**

**{**

**struct p**

**{**

**int x;**

**char y;**

**q ptr;**

**};**

**struct p p = {1, 2, &p};**

**printf("%d\n", p.ptr->x);**

**return 0;**

**}**

**a) Compile time error**

**b) 1**

**c) Depends on the compiler**

**d) None of the mentioned**

**47. What is the output of this C code?**

**#include <stdio.h>**

**int main()**

**{**

**typedef struct p \*q;**

**struct p**

**{**

**int x;**

**char y;**

**q ptr;**

**};**

**struct p p = {1, 2, &p};**

**printf("%d\n", p.ptr->x);**

**return 0;**

**}**

**a) Compile time error**

**b) 1**

**c) Depends on the compiler**

**d) Depends on the standard**

**48. What is the output of this C code?**

**#include <stdio.h>**

**typedef struct p \*q;**

**struct p**

**{**

**int x;**

**char y;**

**q ptr;**

**};**

**int main()**

**{**

**struct p p = {1, 2, &p};**

**printf("%d\n", p.ptr->ptr->x);**

**return 0;**

**}**

**a) Compile time error**

**b) Segmenation fault**

**c) Undefined behaviour**

**d) 1**

**49. The number of distinct nodes the following struct declaration can**

**point to is.**

**struct node**

**{**

**struct node \*left;**

**struct node \*centre;**

**struct node \*right;**

**};**

**a) 1**

**b) 2**

**c) 3**

**d) All of the mentioned**

**50. Which of the following is not possible?**

**a) A structure variable pointing to itself**

**b) A structure variable pointing to another structure variable of**

**same type**

**c) 2 different type of structure variable pointing at each other.**

**d) None of these**

**51. For the following declaration of structure, which of the following**

**will stop the loop at the last node of a linked list?**

**struct node**

**{**

**struct node \*next;**

**};**

**a) while (p != NULL)**

**{**

**p = p->next;**

**}**

**b) while (p->next != NULL)**

**{**

**p = p->next;**

**}**

**c) while (1)**

**{**

**p = p->next;**

**if (p == NULL)**

**break;**

**}**

**d) All of the mentioned**

**52. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*name;**

**};**

**struct student s;**

**struct student fun(void)**

**{**

**s.name = "newton";**

**printf("%s\n", s.name);**

**s.name = "alan";**

**return s;**

**}**

**void main()**

**{**

**struct student m = fun();**

**printf("%s\n", m.name);**

**m.name = "turing";**

**printf("%s\n", s.name);**

**}**

**a) newton alan alan**

**b) alan newton alan**

**c) alan alan newton**

**d) Compile time error**

**53.What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*name;**

**};**

**void main()**

**{**

**struct student s, m;**

**s.name = "st";**

**m = s;**

**printf("%s%s", s.name, m.name);**

**}**

**a) Compile time error**

**b) Nothing**

**c) Junk values**

**d) st st**

**54. Which of the following return-type cannot be used for a function in**

**C?**

**a) char \***

**b) struct**

**c) void**

**d) None of the mentioned**

**55.Which of the following is not possible under any scenario?**

**a) s1 = &s2;**

**b) s1 = s2;**

**c) (\*s1).number = 10;**

**d) None of the mentioned**

**56. Which of the following operation is illegal in structures?**

**a) Typecasting of structure**

**b) Pointer to a variable of same structure**

**c) Dynamic allocation of memory for structure**

**d) All of the mentioned**

**57.Presence of code like “s.t.b = 10? indicate.**

**a) Syntax Error**

**b) structure**

**c) double data type**

**d) An ordinary variable name**

**58. The output of the code below is**

**#include <stdio.h>**

**struct student**

**{**

**char \*name;**

**};**

**struct student fun(void)**

**{**

**struct student s;**

**s.name = "alan";**

**return s;**

**}**

**void main()**

**{**

**struct student m = fun();**

**s.name = "turing";**

**printf("%s", m.name);**

**}**

**a) alan**

**b) Turing**

**c) Compile time error**

**d) Nothing**

**59. What is the output of this C code?**

**#include <stdio.h>**

**struct point**

**{**

**int x;**

**int y;**

**};**

**int main()**

**{**

**struct point p = {1};**

**struct point p1 = {1};**

**if(p == p1)**

**printf("equal\n");**

**else**

**printf("not equal\n");**

**}**

**a) Compile time error**

**b) equal**

**c) depends on the standard**

**d) not equal**

**60. What is the output of this C code?**

**#include <stdio.h>**

**struct point**

**{**

**int x;**

**int y;**

**};**

**struct notpoint**

**{**

**int x;**

**int y;**

**};**

**struct point foo();**

**int main()**

**{**

**struct point p = {1};**

**struct notpoint p1 = {2, 3};**

**p1 = foo();**

**printf("%d\n", p1.x);**

**}**

**struct point foo()**

**{**

**struct point temp = {1, 2};**

**return temp;**

**}**

**a) Compile time error**

**b) 1**

**c) 2**

**d) Undefined behaviour**

**61. What is the output of this C code?**

**#include <stdio.h>**

**struct point**

**{**

**int x;**

**int y;**

**};**

**struct notpoint**

**{**

**int x;**

**int y;**

**};**

**int main()**

**{**

**struct point p = {1};**

**struct notpoint p1 = p;**

**printf("%d\n", p1.x);**

**}**

**a) Compile time error**

**b) 1**

**c) 0**

**d) Undefined**

**62.What is the output of this C code?**

**#include <stdio.h>**

**struct point**

**{**

**int x;**

**int y;**

**};**

**struct notpoint**

**{**

**int x;**

**int y;**

**};**

**void foo(struct point);**

**int main()**

**{**

**struct notpoint p1 = {1, 2};**

**foo(p1);**

**}**

**void foo(struct point p)**

**{**

**printf("%d\n", p.x);**

**}**

**a) Compile time error**

**b) 1**

**c) 0**

**d) Undefined**

**63. What is the output of this C code?**

**#include <stdio.h>**

**struct point**

**{**

**int x;**

**int y;**

**};**

**void foo(struct point\*);**

**int main()**

**{**

**struct point p1 = {1, 2};**

**foo(&p1);**

**}**

**void foo(struct point \*p)**

**{**

**printf("%d\n", \*p.x++);**

**}**

**a) Compile time error**

**b) Segmentation fault/code crash**

**c) 2**

**d) 1**

**64. What is the output of this C code?**

**#include <stdio.h>**

**struct point**

**{**

**int x;**

**int y;**

**};**

**void foo(struct point\*);**

**int main()**

**{**

**struct point p1 = {1, 2};**

**foo(&p1);**

**}**

**void foo(struct point \*p)**

**{**

**printf("%d\n", \*p->x++);**

**}**

**a) Compile time error**

**b) 1**

**c) Segmentation fault/code crash**

**d) 2**

**65. What is the output of this C code?**

**#include <stdio.h>**

**struct student fun(void)**

**{**

**struct student**

**{**

**char \*name;**

**};**

**struct student s;**

**s.name = "alan";**

**return s;**

**}**

**void main()**

**{**

**struct student m = fun();**

**printf("%s", m.name);**

**}**

**a) Compile time error**

**b) alan**

**c) Nothing**

**d) Varies**

**66. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*name;**

**};**

**struct student fun(void)**

**{**

**struct student s;**

**s.name = "alan";**

**return s;**

**}**

**void main()**

**{**

**struct student m = fun();**

**printf("%s", m.name);**

**}**

**a) Nothing**

**b) alan**

**c) Run time error**

**d) Varies**

**67. The correct syntax to access the member of the ith structure in**

**the array of structures is?**

**Assuming: struct temp**

**{**

**int b;**

**}s[50];**

**a) s.b.[i];**

**b) s.[i].b;**

**c) s.b[i];**

**d) s[i].b;**

**68. Comment on the output of this C code?**

**#include <stdio.h>**

**struct temp**

**{**

**int a;**

**int b;**

**int c;**

**};**

**main()**

**{**

**struct temp p[] = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};**

**}**

**a) No Compile time error, generates an array of structure of size 3**

**b) No Compile time error, generates an array of structure of size 9**

**c) Compile time error, illegal declaration of a multidimensional**

**array**

**d) Compile time error, illegal assignment to members of structure**

**69. Which of the following uses structure?**

**a) Array of structures**

**b) Linked Lists**

**c) Binary Tree**

**d) All of the mentioned**

**70. What is the correct syntax to declare a function foo() which**

**receives an array of structure in function?**

**a) void foo(struct \*var);**

**b) void foo(struct \*var[]);**

**c) void foo(struct var);**

**d) None of the mentioned**

**71. What is the output of this C code?**

**(Assuming size of int be 4)**

**#include <stdio.h>**

**struct temp**

**{**

**int a;**

**int b;**

**int c;**

**} p[] = {0};**

**main()**

**{**

**printf("%d", sizeof(p));**

**}**

**a) 4**

**b) 12**

**c) 16**

**d) Can’t be estimated due to ambigous initialization of array**

**72. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*name;**

**};**

**struct student s[2];**

**void main()**

**{**

**s[0].name = "alan";**

**s[1] = s[0];**

**printf("%s%s", s[0].name, s[1].name);**

**s[1].name = "turing";**

**printf("%s%s", s[0].name, s[1].name);**

**}**

**a) alan alan alan turing**

**b) alan alan turing turing**

**c) alan turing alan turing**

**d) Run time error**

**73. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*name;**

**};**

**struct student s[2], r[2];**

**void main()**

**{**

**s[0].name = "alan";**

**s[1] = s[0];**

**r = s;**

**printf("%s%s", r[0].name, r[1].name);**

**}**

**a) alan alan**

**b) Compile time error**

**c) Varies**

**d) Nothing**

**74. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**char \*name;**

**};**

**void main()**

**{**

**struct student s[2], r[2];**

**s[1] = s[0] = "alan";**

**printf("%s%s", s[0].name, s[1].name);**

**}**

**a) alan alan**

**b) Nothing**

**c) Compile time error**

**d) Varies**

**75. What is the output of this C code?**

**#include <stdio.h>**

**struct student**

**{**

**};**

**void main()**

**{**

**struct student s[2];**

**printf("%d", sizeof(s));**

**}**

**a) 2**

**b) 4**

**c) 8**

**d) 0**

**76. What is the output of this C code?**

**#include <stdio.h>**

**struct p**

**{**

**int x;**

**char y;**

**};**

**int main()**

**{**

**struct p p1[] = {1, 92, 3, 94, 5, 96};**

**struct p \*ptr1 = p1;**

**int x = (sizeof(p1) / 3);**

**if (x == sizeof(int) + sizeof(char))**

**printf("%d\n", ptr1->x);**

**else**

**printf("falsen");**

**}**

**a) Compile time error**

**b) 1**

**c) Undefined behaviour**

**d) false**

**77. What is the output of this C code?**

**#include <stdio.h>**

**struct p**

**{**

**int x;**

**char y;**

**};**

**int main()**

**{**

**struct p p1[] = {1, 92, 3, 94, 5, 96};**

**struct p \*ptr1 = p1;**

**int x = (sizeof(p1) / sizeof(ptr1));**

**if (x == 1)**

**printf("%d\n", ptr1->x);**

**else**

**printf("false\n");**

**}**

**a) Compile time error**

**b) 1**

**c) false**

**d) Undefined behaviour**

**78. What is the output of this C code?**

**#include <stdio.h>**

**struct p**

**{**

**int x;**

**char y;**

**};**

**typedef struct p\* q\*;**

**int main()**

**{**

**struct p p1[] = {1, 92, 3, 94, 5, 96};**

**q ptr1 = p1;**

**printf("%d\n", ptr1->x);**

**}**

**a) Compile time error**

**b) 1**

**c) Undefined behaviour**

**d) Segmentation fault**

**79. What is the output of this C code?**

**#include <stdio.h>**

**struct p**

**{**

**int x;**

**char y;**

**};**

**void foo(struct p\* );**

**int main()**

**{**

**typedef struct p\* q;**

**struct p p1[] = {1, 92, 3, 94, 5, 96};**

**foo(p1);**

**}**

**void foo(struct p\* p1)**

**{**

**q ptr1 = p1;**

**printf("%d\n", ptr1->x);**

**}**

**a) Compile time error**

**b) 1**

**c) Segmentation fault**

**d) Undefined behaviour**

**80. Which of the following are incorrect syntax for pointer to**

**structure?**

**(Assuming struct temp{int b;}\*my\_struct;)**

**a) \*my\_struct.b = 10;**

**b) (\*my\_struct).b = 10;**

**c) my\_struct->b = 10;**

**d) Both ( a) and (b)**

**81. Which of the following is a User-defined data type?**

**a) typedef int Boolean;**

**b) typedef enum {Mon, Tue, Wed, Thu, Fri} Workdays;**

**c) struct {char name[10], int age};**

**d) all of the mentioned**

**82. What is the output of this C code?**

**#define MAX 2**

**enum bird {SPARROW = MAX + 1, PARROT = SPARROW + MAX};**

**int main()**

**{**

**enum bird b = PARROT;**

**printf("%d\n", b);**

**return 0;**

**}**

**a) Compilation error**

**b) 5**

**c) Undefined value**

**d) 2**

**83. int main()**

**{**

**enum {ORANGE = 5, MANGO, BANANA = 4, PEACH};**

**printf("PEACH = %d\n", PEACH);**

**}**

**a) PEACH = 3**

**b) PEACH = 4**

**c) PEACH = 5**

**d) PEACH = 6**