一、单选题

1. A 2. C 3. B 4. B 5. B 6. C 7. A 8. D

9. B 10. D 11. D 12. C 13. D 14. D 15. C 16. A

17. B 18. A 19. C 20. B 21. B 22. C 23. C 24. B

25. D 26. C 27. A 28. A 29. C 30. B 31. D 32. C

33. D 34. C 35. C 36. A 37. B 38. B 39. C 40. A

41. C 42. C 43. A 44. B 45. B 46. A 47. B 48. B

49. C 50. B 51. D 52. B 53. A 54. D 55. C 56. D

57. C 58. B 59. C 60. D 61. D 62. B 63. A 64. C

65. A 66. C 67. C 68. B 69. B 70. C 71. A 72. D

73. B 74. D 75. A 76. C 77. D 78. D 79. C 80. C

81. D 82. A 83. B 84. D 85. A 86. A 87. B 88. C

89. D 90. C 91. A 92. D 93. D 94. B 95. D 96. C

97. A 98. B. 99. C 100. D 101. A 102. B 103. B 104. C

105. D 106. B 107. A 108. D 109. B 110. C 111. D 112. A

113. A 114. B 115. D 116. B 117. A 118. C 119. A 120. B

121. D 122. C 123. A 124. C 125. B 126. A 127. C 128. C

129. D 130. B 131. D 132. C 133. A 134. B 135. A 136. C

137. C 138. B 139. A 140. C 141. B 142. A 143. D 144. D

145. B 146. A 147. C 148. D 149. B 150. C 151. D 152. C

153. D 154. B 155. A 156. C 157. D 158. C 159. D 160. A

161. A 162. B 163. D 164. B 165. C 166. B 167. C 168. C

169. D 170. A 171. C 172. D 173. B 174. A 175. D 176. D

177. C 178. A 179. D 180. B 181. B 182. C 183. A 184. C

185. C 186. D 187. C 188. B 189. A 190. C 191. D 192. A

193. A 194. D 195. B 196. C 197. B 198. C 199. B 200. D

201. C 202. B 203. A 204. C 205. B 206. D 207. D 208. A

二、判断题

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **×** | **6** | **×** | **11** | **×** | **16** | **√** | **21** | **√** | **26** | **√** | **31** | **√** | **36** | **√** | **41** | **√** |
| **2** | **√** | **7** | **×** | **12** | **×** | **17** | **×** | **22** | **×** | **27** | **√** | **32** | **×** | **37** | **√** |  |  |
| **3** | **√** | **8** | **×** | **13** | **√** | **18** | **√** | **23** | **√** | **28** | **×** | **33** | **√** | **38** | **√** |  |  |
| **4** | **×** | **9** | **×** | **14** | **√** | **19** | **×** | **24** | **×** | **29** | **×** | **34** | **√** | **39** | **×** |  |  |
| **5** | **√** | **10** | **√** | **15** | **×** | **20** | **×** | **25** | **√** | **30** | **√** | **35** | **√** | **40** | **√** |  |  |

三、填空题

1. C

2. .cpp、程序

3. //

4. /\*、\*/

5. cout

6. cin

7. main

8. 复合

9. 空白

10. #include

11. #

12. 分号

13. 用户（或编程者）

14. 程序

15. 函数头、函数体

16. 原型（或声明）

17. .h、.cpp

18. 警告（warning）、致命(error)

19. void

20. void

21. int0

22.下一行

23. 35

24. 25

25. 6、30

26. 4、11

27. 325

28. 229

29. 0xF5(F大小写均可)

30. 0365

31. -128、+127

32. 44、

33. 4、8

34. 1、1

35. 2、4

36. 15

37. 15

38. 7

39. 6

40. 1、3

41. 3、3.2

42. x

43. true(或1)

44. false(或0)

45. x

46. !x

47. false(或0)

48. true(或1)

49. (x+y)/(2\*x\*y)

50. 3\*x\*y/(x+y)

51. 1/(a\*x\*x+b\*y\*y)

52. 1

53. 3、

54. 4、 2

55. 枚举常量、 整数

56. int、 double

57. 6、 60

58. 4、 1

59. 20

60. 9 216

61. 0、 19

62. 常量

63. if

64. switch

65. 不停止

66. 11

67. 10

68. break

69. continue

70. return

71. 主(或main)

72. 40

73. 192

74. 300

75. 1、 2

76. 0、 1

77. 8

78. 11

79. n+1

80. cin>>a

81. cout<<a

82. 10、 19

83. ”456”

84. 5、1

85. 9、 7

86. 8、 0

87. 函数体

88. 类型

89. static

90. extern double x;

91. template

92. 虚拟类型(或类型)

93. 递归

94. template<class T>T cubin(T n){return n\*n\*n;}

95. void

96. static

97. 右边（或后面） 98. extern

99. 头文件

100. 函数体

101. 文件包含命令

102. 文件

103. 代码区（程序区）

104. 全局数据

105. 栈

106. i\*m+j+1

107. 2、 4

108. 13

109. 总和

110. 4

111. 地址 DataType\*

112. (char\*)p

113. int\*\*

114. \*p p

115. \*p、 p

116. \*p、 \*p

117. 25

118. 42

119. 26

120. 42

121. 4\*i

122. \*(a+i)

123. a+i\*sizeof(a[i])

124. 第一个、  修改

125. b[7]、b[2]

126. int \*p=x;

127. \*p

128. int[n]、int \*

129. c[3][0]、c[3][2]

130. int(\*)[6]  
131. \*(a[i]+j) 或\*(\*(a+i)+j)或\*(a+i)[j]

132. int y=x;   
133. x

134. 相等、 x

135. \*p

136. p[0]、  \*P

137. ’a’

138. char(\*)[n]

139. delete p

140. delete []p

141. 0 （或’\0’）

142. double \*pv=v;

143. H

144. ple

145. pc=(char \*)pv;

146. int (\*p)[100]=b;

147. (\*p).name

148. fp->score

149. int ra=a;

150. 6

151. 10

152. 17

153. 对象

154. 实例

155.封装

156. 继承

157.聚合

158. 抽象、封装体

159.继承

160. 多继承（或多重继承）

161. 单继承

162. 编译、  运行

163. 重载、  虚函数

164. 设计、  实现

165. 问题域、  应用

166. 函数

167. private（或私有）

168. public(或公有)

169. private(或私有)

170. public(或公有)

171. private

172. 内联

173. 类名

174. 冒号

175. inline

176. 作用域

177. AA\* AA::abc()

178. 不在

179. const

180. const

181. \*this

182. this

183. 类定义

184. 不能够

185. 队列长度

186. 队首

187. (first+length)MS

188. (first+1)MS

189. length==0

190. length==MS

191. 删除

192. 插入

193.队首

194. 队尾

195. ElemHead->next

196. ElemTail

197. NULL

198. 相同

199. 相同

200. 2

201. 构造

202. 析构

203. 构造函数

204. AB() {}

205. ~AB() {}

206. 无参构造函数

207. 动态存储空间

208. 析构函数

209. 10

210. N

211. 初始化表

212. 初始化表

213. 初始化表

214. 函数体

215. const int aa;

216. int aa;

217. AB(int aa):a(aa){};

218. AB(int aa):a(aa){};

219. 析构函数

220. 不会

221. 2

222. a

223. 2

224. 1

225. 2

226. 1

227. 3

228. 错误

229. 正确

230. 错误

231. 拷贝构造函数

232. AB(AB)（注：可附带任何一个名字的参数）

233. delete []pa;

234. 只能、不能

235. 初始化

236. 类域

237. 优先级

238. 0、1

239. 1、2

240. 赋值

241. 2

242. 成员函数

243. ostream

244. istream

245. 2

246. 成员

247. 返回值、空

248. 继承（派生）

249. 公有和保护、私有

250. 公有和保护、任何

251. private、私有

252. 类名

253. 虚基类

254. 二义性

255. this

四、程序填充

1. (1) f1+f2 (2) i%5==0 (3) f

2. (1) x (2) i (3) -j

3. (1) n<=99 (2) i<=temp (3) break

4. (1) b<=0 (2) a=b (3) b=r

5. (1) x<3 (或x<=2) (2) x%i==0 (3) i++

6. (1) b%i==0 (2) i++ (或++i) (3) b>=i

7. (1) i<=5 (2) j>=5+I (3) cout<<’\*’

8. (1) str[i]>=’A ’&& str[i]<=’Z’

(2) num++ (3) return num

9. (1) void fun(char ss[]) (2) fun(s) (3) i<n/2

10. (1) a[i] (或a[i]!=’\0’) (2) a[i]!=c (3) '\0'

11. (1) cin>>\*p (2) p++ (或++p) (3) q-- (或--q)

12. (1) x/2 (2) x!=0 (3) cout<<a[k]

13. (1) int n (2) k=j (3) a[k]=x

14. (1) int a[] (或int\* a)

(2) a[j+1]=a[j] (3) break

15. (1) (low+high)/2 (2) return mid (3) low=mid+1

16. (1) i<size (2) table[j+1]=table[j] (3) p

17. (1) i<n (2) new NODE (3) table[i]

18. (1) i<n (2) table[i] (3) L=p

19. (1) NODE\* p=L (2) p=p->next (3) p->data

20. (1) q->next (2) p (3) return p

21. (1) int& ARRAY:: (2) n>=s (3) v[n] (或\*(v+n))

22. (1) ARRAY:: (2) new int[n] (3) v[i]=a[i]

23. (1) int& ARRAY:: (2) n<1 || n>s (3) v[n-1] (或\*(v+n-1))

24. (1) AA::MaxA() (2) x=a[i] (3) return x

25. (1) i<n-1 (或i<=n-2) (b) {x=a[j]; k=j;} (c) a[i]=x

26. (1) AA::InsertA() (2) a[j+1]=a[j] (3) break

27. (1) x.SetA(a,6) (2) x.MaxA() (3) x.PrintA()

28. (1) AA x (2) m=x.MaxA() (3) x.SortA()

29. (1) top==ARRAY\_SIZE-1 (2) top++（或++top） (3) newElem

30. (1) int Stack::Pop() (2) elem[top--] (3) Pop()

31. 答案：(1) A(int aa):a(aa)

(2) A \*p

(3) p=new A(5)

32. 答案：(1) A() {a=0;} 或A():a(0){} 注：数据0可用NULL代替

(2) new char[strlen(aa)+1]

(3) ~A() {delete []a;}

33. 答案：(1) :a(aa),b(bb)

(2) A x(5),y(x) 注：x(5)与x=5等效，y(x) 与y=x等效

(3) A \*p=&x

34. 答案：(1) A(int aa=0, int bb=0){a=aa; b=bb;}

(2) p1=new A

(3) p2= new A(4,5)

35. 答案：(1) a=new int[MaxLen]

(b) {return a[i];}

(c) s+=r.GetValue(i)

36. 答案：(1) for(int i=0; i<n; i++) a[i]=aa[i]

(2) A::~A() {delete []a;}

(3) for(i=0; i<10; i++) s+=r.GetValue(i)

37. (1) return gd\_name (2) static int (3) return totalweight

38. (1) friend Point operator+(Point& pt,int dd)

(2) friend Point operator+(Point& pt1,Point& pt2)

(3) temp.y+=pt2.y

39. (1) public Base1, public Base2

(2) Base1::x=x1

(3) Base2::x=x2

40. (1) Base::output() (2) Base(m1,m2) (3) mem3=m3

41. (1) Dx()\*Dx()+Dy()\*Dy()

(2) {pt1=pts;pt2=pte;} (或 :pt1(pts),pt2(pte){})

(3) Length()

42. (1) virtual public vehicle

(2) virtual public vehicle

(3) public bicycle, public motorcar

五、读程序写出运行结果

1. max:85

min:18

2. 1 2 3 5 8 13 21 34

55 89

3. s=60

4. 121

a,b,c=7,12,30

5. 11 14 switch end.

6. 36,139,-5,1,73,8 ,8,67

7. 114 138 55

8. 1 1 1

2 2 3

3 6 9

4 24 33

9. 10 6 4

10. \*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

11. 1 5 7 11 13 37

12. ++\*+\*\*\*\*\*

13. 1 2 12

2 2 30

3 2 66

14. s=63

15. 264

16. 10 1 -6

4 0 -3

6 -6 -9

17. 10 25

15 35

10 25

18. x=13, y=21

x=5, y=8, z=34

19. x,y=10, 26

x,y=26, 10

x,y=10, 26

x,y=25, 11

20. 29 28 57

21. 55 50

22. 4 3 14

23. 2 1 5 2

24. worker cadre

25. 5 8 11 14

26. +-\*/ abcd 1234

27. year:4

month:5

day:3

28. abcdef 30

shenyafen 95

29. abcdef 30

abcdefxyz 50

30. 12

20

31. 4

27

32. a: 7/12

b: -3/8

c: 5/24

d: 19/24

33. a: 6/15

b: 3/10

c: 7/10

d: 11/10

34. xuxiaokai

weirong

Destructor!

Destructor!

35. 15 8

36. 1 2 3 4

0 0 0

XxkXxkXxk

37. 5 6

9 20

38. d=800

f=40

39. 14 6 40 2.5

@非法运算符!

40. Point:3 4

Circle radius:5

Circle destructor!

Point destructor!

41. Point:3 4

Text con!

Point with Text con!

Point with Text des

Text des!

Point des!

42. 2000/1/1

0:0:0

2002/10/1

23:59:59

2002/12/31

0:0:0

43. 2001/1/1

5:30:0

2002/10/1

6:20:0

2003/3/8

6:7:0

44. Base(1,2)

Next(10,15)

Base(5,7)

45. 10,20,30,40

15,26,30,40

15,26,210,280

46. B\_obj x=100

D\_obj y=200

B\_obj x=300

D\_obj y=400

六、指出以下程序或函数的功能

1. 计算并输出22+42+62++302的值。

2. 随机产生出10个10至100之间的素数并显示出来。

3. 此函数用于把十进制整数x转换为十六进制数字串输出

4. 计算并输出1+的值，其中a的值由键盘输入。

5. 以参数a和b为运算对象，以参数op为四则算术运算符，求出运算结果并返回。

6. 把从键盘上输入的一个整数x分解为所有可能的每两个因子之积。

7. 计算并输出的值，其中N值由键盘输入。

8. 让计算机随机产生出10道加数在20以内整数的加法题供用户计算，每道题10分，计算完成后打印出得分。

9. 用递归求出1+22+32+...+n2的值。

10. 实现字符串系统函数strcpy的功能，把b所指的字符串拷贝到a所指的字符数组空间中。

11. 函数模板，从一维数组a[n]中查找值为key的元素，若查找成功则返回真否则返回假。

12. 显示输出数组a[n]中大于等于平均值的所有元素值。

13. 对于二维字符数组a中保存的M个字符串，分别统计并输出其长度小于5、大于等于5且小于15、大于等于15的字符串个数。

14. 从一个二维整型数组中查找具有最大值的元素，由引用参数row和col带回该元素的行号和列号。

15. 一个递归函数过程，求出两个自然数m和n的最小公倍数。

16. 实现strcat函数的功能，把str2所指字符串连接到str1所指字符串的后面，并返回str1指针。

17. 实现strcmp函数的功能，比较两个字符串str1和str2的大小，若str1较大则返回1，若str2较大则返回-1，若两者相等则返回0。

18. 从表头指针f指向的、由IntNode类型的结点所构成的链表中查找出data域的值最大的结点并返回指向该结点的指针。

19. 统计出以表头指针为f的链表中结点的个数。

20. 对于以表头指针为f的链表，依次显示出每个结点的data域的值。

21. 首先从键盘上输入一个整数给n，然后依次输入n个整数建立以表头指针为f的链表。

22. 求出并返回字符指针参数s所指向的字符串长度。

23. 根据参数s所指向的字符串，生成一个由r所指向的新字符串并返回，该字符串使s字符串中的小写字母均变为大写。

七、程序改错

1. 2 8

NODE \*p=new NODE; p1->next=p;

2. 3 5

p=new NODE; p->next=plist;

3. 5 7 8

b.InitFranction(1,3); c=a.FranAdd(b); c.FranOutput()

4. 6 9 10

b->InitFranction(3,4); c=a->FranAdd(\*b); c.FranOutput()

5. 5 9 12

public: } int CE::GetMin() {

6. 4 5 6

Public:

A():c(0),a(0),b(0) {}

A(int aa, int bb): c(aa+bb) {a=aa; b=bb;}

7. 5 8 9

在函数体给常量c赋值

定义b多一个参数

定义y少一个参数

8. 13 15 17

在函数体给常量c赋值 缺少分号 定义z多一个参数，分号前多一个逗号

9. 5 19 20

a[i]=aa[i]; d\*=x.Get(i); int f=x.SumA(6);

10. 2 5

friend fract &operator+=(fract&,fract);

fract &operator+=(fract &f1,fract f2);

八、程序设计

1. 若采用for循环编写程序，则如下所示：

void main()

{

int i=0; double s=0;

while(s<5) s+=double(1)/++i;

cout<<"n="<<i<<endl;

}

#include<iostream.h>

void main()

{

int i; double s=0;

for(i=1; s<5; i++) s+=1.0/i;

cout<<"n="<<i-1<<endl;

//注意：此i-1的值为所求的n值

}

2. #include<iostream.h>

void main()

{

int i; //用i作为循环变量

int p=1; //用p作为累乘变量

int s=1; //用s作为累加循环变量

for(i=1;i<=10;i++) {p\*=3; s+=p;}

cout<<s<<endl;

}

3. #include<iostream.h>

void main()

{

int i=0; //用i作为依次取值偶数的变量

int s=0; //用s作为累加变量

do {

i+=2;

s+=i\*i;

}while(s<1000);

cout<<"n="<<i-2<<endl;

}

4. #include<iomanip.h>

#include<math.h>

const double a=10.2;

void main()

{

double x,y;

cin>>x;

while(x!=-100) {

if(x<=0) y=sqrt(a\*a+x\*x);

else y=3\*a\*x\*x+4\*a\*x-1;

cout<<x<<’ ’<<y<<endl;

cin>>x;

}

}

5. #include<iostream.h>

void main()

{

int x,max;

cin>>x;

max=x;

for(int i=0;i<9;i++) {

cin>>x;

if(x>max) max=x;

}

cout<<"max:"<<max<<endl;

}

6. #include<iostream.h>

void main()

{

int a,b;

for(a=6;a<=30; a++)

for(b=15;b<=36;b++)

if(2\*a+5\*b==126)

cout<<'('<<a<<','<<b<<')'<<endl;

}

7. #include<iomanip.h>

void main()

{

int c1,c2,c3;

double x;

c1=c2=c3=0;

cin>>x;

while(x>0) {

if(x<=12) c1++;

else if(x<=15) c2++;

else c3++;

cin>>x;

}

cout<<c1<<’ ’<<c2<<’ ’<<c3<<endl;

}

8. void fun4(char\* a, int b[])

{

int i;

for(i=0;i<10;i++) b[i]=0;

while(\*a) {

int j=\*a-’0’;

if(j>=0 && j<=9) b[j]++;

a++;

}

}

9. double Mean(double a[M][N],int m,int n);

double Mean(double a[M][N],int m,int n)

{

int i,j;

double v=0.0;

for(i=0; i<m; i++)

for(j=0; j<n; j++)

v+=a[i][j];

v/=m\*n;

return v;

}

10.int f(int a[],int n)

{

if(n==0) return 0;

else return a[n-1]\*a[n-1]+f(a,n-1);

}

11.

void p(int n)

{

if(n!=0) {

for(int i=0; i<n; i++) cout<<n;

cout<<endl;

p(n-1);

}

}

12. void p(int n)

{

if(n!=0) {

p(n-1);

for(int i=0; i<n; i++) cout<<n;

cout<<endl;

}

}

13.

int AA::Count(int x)

{

int i,c=0;

for(i=0; i<n;i++)

if(a[i]==x) c++;

return c;

}

14.

int AA::Search(int x)

{

int i;

for(i=0; i<n;i++)

if(a[i]==x) return i;

return -1;

}

15.

int AA::MaxMin(int& x, int& y)

{

int mx,my;

mx=my=a[0];

for(int i=1; i<n; i++) {

if(a[i]>mx) mx=a[i];

if(a[i]<my) my=a[i];

}

x=mx; y=my;

if(n>0) return 1; else return 0;

}

16.

int AA::Compare(AA b)

{

if(n!=b.n) return 0;

for(int i=0; i<n; i++)

if(a[i]!=b.a[i]) return 0;

return 1;

}

17.

int AA::CompareBig(AA b)

{

int k;

if(n>b.n) k=b.n; else k=n;

for(int i=0; i<k; i++)

if(a[i]>b.a[i]) return 1;

else if(a[i]<b.a[i]) return -1;

if(k==n && k==b.n) return 0;

else if(k<n) return 1;

else return -1;

}

18.

AA\* AA::Reverse()

{

int i,x;

for(i=0; i<n/2; i++) {

x=a[i]; a[i]=a[n-1-i]; a[n-1-i]=x;

}

return this;

}

19.

AA\* AA::Reverse1()

{

AA \*px=new AA;

px->n=n; px->MS=MS;

px->a=new int[MS];

for(int i=0; i<n; i++) {

px->a[n-1-i]=a[i];

}

return px;

}

20.

Array::Array(int aa[], int nn) {

n=nn;

a=new int[n];

for(int i=0; i<n; i++) a[i]=aa[i];

}

21.

Array::Array(Array& aa) {

n=aa.n;

a=new int[n];

for(int i=0; i<n; i++) a[i]=aa.a[i];

}

22.

Array& Array::Give(Array& aa) {

delete []a;

n=aa.n;

a=new int[n];

for(int i=0; i<n; i++) a[i]=aa.a[i];

return \*this;

}

23.

Array Array::Uion(Array& aa) {

int k=n+aa.n;

int\* w=new int[k];

int i;

for(i=0; i<n; i++) w[i]=a[i];

for(i=0; i<aa.n; i++) w[i+n]=aa.a[i];

Array r(w,k);

return r;

}

24.

Strings::Strings(char\*str) {

n=strlen(str);

s=new char[n+1];

strcpy(s,str);

}

25.

Strings::Strings(Strings& str) {

n=str.n;

s=new char[n+1];

strcpy(s,str.s);

}

26.

Strings& Strings::Give(Strings& str) {

delete []s;

n=str.n;

s=new char[n+1];

strcpy(s,str.s);

return \*this;

}

27.

Strings Strings::Uion(Strings& str) {

Strings r("");

r.n=n+str.n;

r.s=new char[r.n+1];

strcpy(r.s,s);

strcat(r.s,str.s);

return r;

}

28.

class A\_class {

private:

int x,y,t;

public:

A\_class(int i,int j):x(i),y(j){

if(y>x){t=x;x=y;y=t;};

}

//其它函数从略

friend int Variance (A\_class& ac); //参数名任意

};

int Variance (A\_class& ac)

{

return ac.x\*ac.x-ac.y\*ac.y;

}

void main() {

A\_class A\_obj(3,5);

cout<<"Result:"<<Variance(A\_obj)<<endl;

}

29.

for(int i=0; i<m1.rows; i++)

for(int j=0; j<m2.cols; j++){

sum=0;

for(int k=0; k<m1.cols; k++)

sum+=m1.mem[i\*m1.cols+k]\*m2.mem[k\*m2.cols+j];

m3.mem[i\*m3.cols+j] = sum;

}

return true;

30.

Rectangle (float len,float wid):length(len),width(wid){}

~Rectangle (){}

float GetPerimeter() {return 2\*(length+width);} //计算矩形周长

float GetAre() {return length\*width;} //计算矩形面积

31.

Circle(float rad):rad(rad){}

~Circle(){}

float GetPerimeter() {return 2\*PI\*rad;} //计算圆形周长

float GetAre() {return PI\*rad \*rad;} //计算圆形面积