

```

1  /*Bai 1:
2  Chuong trinh nhap vao 1 day so thuc.
3  ->Sap xep day tang sau do in ra man hinh
4  */
5  #include <stdio.h>
6  #include <conio.h>
7  #include <stdlib.h>
8
9  void NhapMang(float *pa, int *pn);
10 void XuatMang(float *pa, int *pn);
11 void HoanVi(float *pa, float *pb);
12 void SapXepTang(float *pa, int *pn);
13
14 void main()
15 {
16     float *pa; //Khai bao con tro pa chi den mang a
17     int *pn ; //Khai bao con tro pn chi den bien n
18     pa = (float*) calloc(100,sizeof(float)); //Khoi tao vung nho chua 100 bien float
19     pn = (int*) malloc(sizeof(int));          //Khoi tao vung nho chua 1 bien int
20     clrscr();
21     //Goi ham nhap mang:
22     NhapMang(pa,pn);
23     //Xuat mang da nhap
24     printf("\nMang da nhap la:");
25     XuatMang(pa,pn);
26     //Sap xep mang da nhap
27     SapXepTang(pa,pn);
28     //Goi ham xuat mang sau khi da sap xep
29     printf("\nMang sau khi sap xep tang la:");
30     XuatMang(pa,pn);
31     getch();
32 }
33 //Ham nhap mang su dung con tro
34 void NhapMang(float* pa, int *pn)
35 {
36     int i;
37     printf("\nNhap vao so phan tu cua mang:");
38     scanf("%d",pn);
39
40     for(i = 0; i < *pn; i++) //lay gia tri cua vung nho ma con tro pn dang tro toi
41     {
42         printf("a[%d] = ",i);
43         scanf("%f",pa+i);    //luu gia tri tai dia chi thu i cua mang
44     }
45 }
46 //Ham xuat mang su dung con tro
47 void XuatMang(float *pa, int *pn)
48 {
49     int i;
50     for(i = 0; i < *pn; i++)
51     {
52         printf("%.3f",*(pa+i));    //xuat gia tri tai vi tri i cua mang
53     }
54 }
55 //Ham hoan vi 2 so thuc
56 void HoanVi(float *a, float *b)

```

```

57 {
58     float t;
59     t = *a; /*a: lay gia tri tai o nho ma con tro a dang tro toi
60     *a = *b; /*b: lay gia tri tai o nho ma con tro b dang tro toi
61     *b = t;
62 }
63 //Ham sap xep day so tang
64 void SapXepTang(float *pa, int *pn)
65 {
66     int i,j;
67     for(i = 0; i < (*pn - 1); i++)
68         for(j = i+1; j < *pn; j++)
69             {
70                 if(*(pa+i) > *(pa+j)) //Neu so dung truoc lon hon so dung sau thi hoan vi
71                     HoanVi(pa+i,pa+j);
72             }
73 }
74
75 /*****
76 *****/
77 /*Bai 2:
78 Chuong trinh sap xep 1 mang theo thu tang dan
79 sau khi loai bo cac phan tu trung nhau !
80 */
81 #include <stdio.h>
82 #include <conio.h>
83 #include <stdlib.h>
84
85 //Khai bao nguyen mau ham
86 void NhapMang(int *pa, int *pn);
87 void XuatMang(int *pa, int *pn);
88 void HoanVi(int *pa, int *pb);
89 void SapXepTang(int *pa, int *pn);
90 void RemoveExist(int *pa, int *pn);
91
92 void main()
93 {
94     int *pa; //Khai bao con tro pa chi den mang a
95     int *pn ; //Khai bao con tro pn chi den bien n
96     pa = (int*) calloc(100,sizeof(int)); //Khoi tao vung nho chua 100 bien int
97     pn = (int*) malloc(sizeof(int)); //Khoi tao vung nho chua 1 bien int
98     clrscr();
99     //Goi ham nhap mang:
100     NhapMang(pa,pn);
101     //Xuat mang da nhap
102     printf("\nMang da nhap la:");
103     XuatMang(pa,pn);
104     //Xoa cac phan tu trung nhau trong mang
105     RemoveExist(pa,pn);
106     //Goi ham xuat mang sau khi da xoa cac phan tu trung nhau
107     printf("\nMang sau khi xoa cac phan tu trung nhau:");
108     XuatMang(pa,pn);
109     //Sap xep mang da nhap
110     SapXepTang(pa,pn);
111     //Goi ham xuat mang sau khi da sap xep
112     printf("\nMang sau khi sap xep tang la:");

```

```

113     XuatMang(pa,pn);
114     getch();
115 }
116 //Ham nhap mang su dung con tro
117 void NhapMang(int* pa, int *pn)
118 {
119     int i;
120     printf("\nNhap vao so phan tu cua mang:");
121     scanf("%d",pn);
122
123     for(i = 0; i < *pn; i++) //lay gia tri cua vung nho ma con tro pn dang tro toi
124     {
125         printf("a[%d] = ",i);
126         scanf("%d",pa+i);    //luu gia tri tai dia chi thu i cua mang
127     }
128 }
129 //Ham xuat mang su dung con tro
130 void XuatMang(int *pa, int *pn)
131 {
132     int i;
133     for(i = 0; i < *pn; i++)
134     {
135         printf("%3d",*(pa+i)); //xuat gia tri tai vi tri i cua mang
136     }
137 }
138 //Ham hoan vi 2 so nguyen
139 void HoanVi(int *a, int *b)
140 {
141     int t;
142     t = *a; //a: lay gia tri tai o nho ma con tro a dang tro toi
143     *a = *b; //b: lay gia tri tai o nho ma con tro b dang tro toi
144     *b = t;
145 }
146 //Ham sap xep day so tang
147 void SapXepTang(int *pa, int *pn)
148 {
149     int i,j;
150     for(i = 0; i < (*pn - 1); i++)
151         for(j = i+1; j < *pn; j++)
152         {
153             if(*(pa+i) > *(pa+j)) //Neu so dung truoc lon hon so dung sau thi hoan vi
154                 HoanVi(pa+i,pa+j);
155         }
156 }
157
158 //Ham loai bo cac phan tu trung nhau trong mang
159 void RemoveExist(int *pa, int *pn)
160 {
161     int i,j,k;
162     for(i = 0; i < *pn-1; i++)
163         for(j = i+1; j < *pn; j++)
164         {
165             if(*(pa+i) == *(pa+j)) //Neu co 2 phan tu bang nhau
166             {
167                 for(k = j; k < *pn-1; k++)
168                     *(pa+k) = *(pa+k+1);

```

```

169         (*pn)--;
170         j--;
171     }
172
173 }
174 }
175 /*****
176 *****/
177 /*Bai 3:
178 Chuong trinh nhap vao mot mang
179 -Tim so lon nhat trong mang
180 -Phan tu nho nhat trong mang
181 -Tinh tong cac phan tu trong mang
182 */
183
184 //Khai bao thu vien
185 #include <stdio.h>
186 #include <conio.h>
187 #include <stdlib.h>
188
189 //Khai bao nguyen mau ham
190 void NhapMang(int *pa, int *pn);
191 void XuatMang(int *pa, int *pn);
192 int TimMax(int *pa, int *pn);
193 int TimMin(int *pa, int *pn);
194 int TinhTong(int *pa, int *pn);
195
196 void main()
197 {
198     int *pa; //Khai bao con tro pa chi den mang a
199     int *pn ; //Khai bao con tro pn chi den bien n
200     pa = (int*) calloc(100,sizeof(int)); //Khoi tao vung nho chua 100 bien int
201     pn = (int*) malloc(sizeof(int));      //Khoi tao vung nho chua 1 bien int
202     clrscr();
203     //Goi ham nhap mang:
204     NhapMang(pa,pn);
205     //Xuat mang da nhap
206     printf("\nMang da nhap la:");
207     XuatMang(pa,pn);
208     printf("\nPhan tu lon nhat trong mang la: %d",TimMax(pa,pn));
209     printf("\nPhan tu nho nhat trong mang la: %d",TimMin(pa,pn));
210     getch();
211 }
212
213 //Ham nhap mang su dung con tro
214 void NhapMang(int* pa, int *pn)
215 {
216     int i;
217     printf("\nNhap vao so phan tu cua mang:");
218     scanf("%d",pn);
219
220     for(i = 0; i < *pn; i++) //lay gia tri cua vung nho ma con tro pn dang tro toi
221     {
222         printf("a[%d] = ",i);
223         scanf("%d",pa+i);    //luu gia tri tai dia chi thu i cua mang
224     }

```

```

225 }
226 //Ham xuất mảng su dung con tro
227 void XuatMang(int *pa, int *pn)
228 {
229     int i;
230     for(i = 0; i < *pn; i++)
231     {
232         printf("%3d",*(pa+i)); //xuất giá trị tại vị trí i của mảng
233     }
234 }
235 //Ham tìm phần tử lớn nhất trong mảng
236 int TimMax(int *pa, int *pn)
237 {
238     int max = *pa+0;//Giá trị max là pt đầu tiên trong mảng
239     int i;
240     for(i = 1; i < *pn; i++)
241         if(max < *(pa+i))
242             max = *(pa+i);
243     return max;
244 }
245 //Ham tìm phần tử nhỏ nhất trong mảng
246 int TimMin(int *pa, int *pn)
247 {
248     int min = *pa+0;//Giá trị min là pt đầu tiên trong mảng
249     int i;
250     for(i = 1; i < *pn; i++)
251         if(min > *(pa+i))
252             min = *(pa+i);
253     return min;
254 }
255 /*****
256 *****/
257 /*Bai 4:
258 Chương trình nhập vào 1 mảng tang
259 - Nếu người dùng nhập sai quy tắc thì nhập lại !
260 - Thêm 1 phần tử vào trong mảng sao cho vẫn đảm bảo nguyên tắc: mảng tang
261 */
262 //Khai báo thư viện
263 #include <stdio.h>
264 #include <conio.h>
265 #include <stdlib.h>
266
267 //Khai báo nguyên mẫu hàm
268 void NhapMang(int *pa, int *pn);
269 void XuatMang(int *pa, int *pn);
270 int KiemTraMangTang(int *pa, int *pn);
271 void ChenPhanTu(int *pa, int *pn, int x);
272
273 void main()
274 {
275     int *pa; //Khai báo con trỏ pa chỉ đến mảng a
276     int *pn ; //Khai báo con trỏ pn chỉ đến biến n
277     int x; //lưu giá trị cần thêm vào trong mảng
278     pa = (int*) calloc(100,sizeof(int)); //Khởi tạo vùng nhớ chứa 100 biến int
279     pn = (int*) malloc(sizeof(int)); //Khởi tạo vùng nhớ chứa 1 biến int
280     clrscr();

```

```

281 //Goi ham nhap mang:
282 NhapMang(pa,pn);
283 while(KiemTraMangTang(pa,pn)==-1)
284 {
285     printf("\nNhap sai nguyen tac ! Nhap lai.");
286     getch();
287     clrscr();
288     NhapMang(pa,pn);
289 }
290 printf("\nMang da nhap la:");
291 XuatMang(pa,pn);
292 printf("\nNhap gia tri muon chen:");
293 scanf("%d",&x);
294 ChenPhanTu(pa,pn,x);
295 printf("\nMang sau khi chen phan tu %d vao mang la:",x);
296 XuatMang(pa,pn);
297 getch();
298 }
299 //Ham nhap mang su dung con tro
300 void NhapMang(int* pa, int *pn)
301 {
302     int i;
303     printf("\nNhap vao so phan tu cua mang:");
304     scanf("%d",pn);
305
306     for(i = 0; i < *pn; i++) //lay gia tri cua vung nho ma con tro pn dang tro toi
307     {
308         printf("a[%d] = ",i);
309         scanf("%d",pa+i); //luu gia tri tai dia chi thu i cua mang
310     }
311 }
312 //Ham xuat mang su dung con tro
313 void XuatMang(int *pa, int *pn)
314 {
315     int i;
316     for(i = 0; i < *pn; i++)
317     {
318         printf("%3d",*(pa+i)); //xuat gia tri tai vi tri i cua mang
319     }
320 }
321 //Ham kiem tra mang tang
322 //Tra ve 0 neu mang tang
323 //Tra ve -1 neu mang khong tang
324 int KiemTraMangTang(int* pa, int *pn)
325 {
326     int i;
327     for(i = 0; i < *pn-1; i++)//Dong for chay tu phan tu dau tien den pt gan cuoi
328         if(*(pa+i) > *(pa+i+1)) //Neu pt dung sau nho hon pt dung truoc
329             return -1; //Tra ve ma -1
330     return 0;
331 }
332 //Ham chen 1 gia tri vao mang sao cho van dam bao nguyen tac: mang tang
333 void ChenPhanTu(int* pa, int *pn, int x)
334 {
335     int i,j;
336     //Truong hop mang 0 co phan tu nao

```

```

337     if(*pn == 0)
338     {
339         *(pa+0) = x;
340         (*pn)++;
341         return; //Thoat khoi chuong trinh
342     }
343     //Truong hop mang chi co duy nhat mot phan tu
344     if(*pn == 1 && x > *(pa+0))
345     {
346         *(pa+1) =x;
347         (*pn)++;
348         return ; //Thoat khoi chuong trinh
349     }
350     //Cac truong hop con lai:
351     for(i = 0; i < *pn; i++)
352     {
353         if(x < *(pa+i))
354         {
355             for(j = *pn-1; j >= i; j--) //Chay tu cuoi mang den vi tri i dang xet
356                 *(pa+j+1) = *(pa+j);
357             //Sau khi day cac phan tu lon hon lai phia sau
358             //Chen x vao vi tri hien tai
359             *(pa+i) = x;
360             (*pn)++;
361             break; //Thoat khoi vong lap
362         }
363     }
364 }
365 /*****
366 *****/
367 /*Bai 5:
368 Chuong trinh nhap vao mot ma tran (mang hai chieu) cac so nguyen.
369 Nhap mot so nguyen khac vao va xe xem co phan tu nao cua ma tran trung
370 voi so nay hay ko? O vi tri nao ? Co bao nhieu phan tu !
371 */
372 #include <stdio.h>
373 #include <conio.h>
374 #include <stdlib.h>
375
376 //Khai bao nguyen mau ham:
377 void NhapMaTran(int *pa, int *pn,int *pm);
378 void XuatMaTran(int *pa, int *pn,int *pm);
379 void KiemTraTonTai(int *pa, int *pn,int *pm, int x);
380
381 void main()
382 {
383     int *pa; //Khai bao con tro pa chi den ma tran a
384     int *pn ; //Khai bao con tro pn chi den bien n(dong)
385     int *pm ; //Khai bao con tro pn chi den bien m(cot)
386     pa = (int*) calloc(100*100,sizeof(int)); //Khoi tao vung nho chua 10000 bien int
387     pn = (int*) malloc(sizeof(int));          //Khoi tao vung nho chua 1 bien int
388     pm = (int*) malloc(sizeof(int));          //Khoi tao vung nho chua 1 bien int
389     int x;  //gia tri do nguoi dung nhap vao
390     clrscr();
391     //Goi ham nhap ma tran
392     NhapMaTran(pa,pn,pm);

```

```

393     //Goi ham xuat ma tran
394     printf("\nMa tran da nhap la:");
395     XuatMaTran(pa,pn,pm);
396     printf("\nNhap vao gia tri x can tim:");
397     scanf("%d",&x);
398     //Tim vi tri cua x trong ma tran:
399     KiemTraTonTai(pa,pn,pm,x);
400     getch();
401 }
402 //Ham nhap ma tran su dung con tro
403 void NhapMaTran(int *pa, int *pn,int *pm)
404 {
405     printf("\nNhap vao so dong:");
406     scanf("%d",pn);    //pn chinh la dia chi cua bien n
407     printf("\nNhap vao so cot:");
408     scanf("%d",pm);    //pm chinh la dia chi cua bien m
409     for(int i= 0; i < *pn; i++)
410         for(int j = 0; j < *pm; j++)
411         {
412             printf("\nx[%d][%d]=",i,j);
413             scanf("%d",(pa+i*(*pm)+j));
414         }
415
416 }
417 //Ham xuat ma tran su dung con tro
418 void XuatMaTran(int *pa, int *pn,int *pm)
419 {
420     int i,j;
421     for(i= 0; i < *pn; i++)
422     {
423         printf("\n");
424         for(j = 0; j < *pm; j++)
425         {
426             printf("%3d",*(pa+i*(*pm)+j)) ;
427         }
428     }
429 }
430 //Ham kiem tra 1 phan tu moi nhap vao da co trong ma tran hay chua ?
431 void KiemTraTonTai(int *pa, int *pn,int *pm, int x)
432 {
433     int i,j;
434     int count = 0; //So phan tu trung voi x
435     for(i = 0; i < *pn; i++)
436         for(j = 0; j < *pm; j++)
437         {
438             if(*(pa+i*(*pm)+j) == x) //Co phan tu trung
439             {
440                 count++; //Tang bien dem len 1
441                 printf("\nTim thay %d tai vi tri: dong(%d) - cot(%d)",x,i,j);
442             }
443         }
444     //Neu count != 0 co nghia la tim thay vi tri trung cua x trong ma tran
445     if(count == 0)
446         printf("\nKhong tim thay %d trong ma tran !");
447     else
448         printf("\nSo lan tim thay %d trong ma tran la: %d",x,count);

```



```

449 }
450 /*****
451 *****/
452 /*Bai 6:
453 Chuong trinh chuyen doi vi tri tu dong thanh cot cua ma tran m*n
454 */
455 #include <stdio.h>
456 #include <conio.h>
457 #include <stdlib.h>
458
459 void NhapMaTran(int *pa, int *pn,int *pm);
460 void XuatMaTran(int *pa, int *pn,int *pm);
461 void ChuyenVi(int *pa, int *pn,int *pm);
462
463 void main()
464 {
465     int *pa; //Khai bao con tro pa chi den ma tran a
466     int *pn ; //Khai bao con tro pn chi den bien n(dong)
467     int *pm ; //Khai bao con tro pn chi den bien m(cot)
468     pa = (int*) calloc(100*100,sizeof(int)); //Khoi tao vung nho chua 10000 bien int
469     pn = (int*) malloc(sizeof(int)); //Khoi tao vung nho chua 1 bien int
470     pm = (int*) malloc(sizeof(int)); //Khoi tao vung nho chua 1 bien int
471     clrscr();
472     NhapMaTran(pa,pn,pm);
473     printf("\nMa tran da nhap la:");
474     XuatMaTran(pa,pn,pm);
475     printf("\nMa tran chuyen vi la:");
476     ChuyenVi(pa,pn,pm);
477     XuatMaTran(pa,pn,pm);
478     getch();
479 }
480
481 //Ham nhap ma tran su dung con tro
482 void NhapMaTran(int *pa, int *pn,int *pm)
483 {
484     printf("\nNhap vao so dong:");
485     scanf("%d",pn); //pn chinh la dia chi cua bien n
486     printf("\nNhap vao so cot:");
487     scanf("%d",pm); //pm chinh la dia chi cua bien m
488     for(int i= 0; i < *pn; i++)
489         for(int j = 0; j < *pm; j++)
490             {
491                 printf("\nx[%d][%d]=",i,j);
492                 scanf("%d",(pa+i*(*pm)+j));
493             }
494
495 }
496 //Ham xuat ma tran su dung con tro
497 void XuatMaTran(int *pa, int *pn,int *pm)
498 {
499     int i,j;
500     for(i= 0; i < *pn; i++)
501     {
502         printf("\n");
503         for(j = 0; j < *pm; j++)
504             {

```

```

505         printf("%3d",*(pa+i*( *pm)+j)) ;
506     }
507 }
508 }
509
510 //Ham hoan vi 2 so nguyen
511 void HoanVi(int *a, int *b)
512 {
513     int t;
514     t = *a; //a: lay gia tri tai o nho ma con tro a dang tro toi
515     *a = *b; //b: lay gia tri tai o nho ma con tro b dang tro toi
516     *b = t;
517 }
518
519 void ChuyenVi(int *pa, int *pn,int *pm)//Sau khi chuyen vi: dong->cot,cot->dong
520 {
521     int x; //Chieu cua ma tran vuong
522     int i,j;
523     int t;
524     x = (*pn > *pm ? *pn : *pm); //x la chieu lon hon
525     for(i = 0; i < x; i++)
526         for(j = i+ 1; j < x; j++)
527             HoanVi((pa+i*( *pm)+j),(pa+j*( *pm)+i));
528     //Luc nay dong thanh cot va cot thanh dong
529     HoanVi(pn,pm);
530 }
531 /*****
532 *****/
533 /*Bai 7:
534 Chuong trinh nhap vao mot mang so tu nhien. Hay xuat ra man hinh:
535 - Dong 1: gom cac so le, tong cong co bao nhieu so le.
536 - Dong 2: gom cac so chan, tong cong co bao nhieu so chan.
537 - Dong 3: gom cac so nguyen to.
538 - Dong 4: gom cac so khong phai la so nguyen to.
539 */
540 #include <stdio.h>
541 #include <conio.h>
542 #include <stdlib.h>
543
544 //Khai bao nguyen mau ham
545 void NhapMang(int *pa, int *pn);
546 void XuatMang(int *pa, int *pn);
547 void HienThiLe(int *pa, int *pn);
548 void HienThiChan(int *pa, int *pn);
549 void LaNguyenTo(int *pa, int *pn);
550 void KhongLaNguyenTo(int *pa, int *pn);
551
552 void main()
553 {
554     int *pa; //Khai bao con tro pa chi den mang a
555     int *pn ; //Khai bao con tro pn chi den bien n
556     pa = (int*) calloc(100,sizeof(int)); //Khoi tao vung nho chua 100 bien int
557     pn = (int*) malloc(sizeof(int)); //Khoi tao vung nho chua 1 bien int
558     clrscr();
559     //Goi ham nhap mang:
560     NhapMang(pa,pn);

```

```

561     //Xuat mang da nhap
562     printf("\nMang da nhap la:");
563     XuatMang(pa,pn);
564     HienThiLe(pa,pn);
565     HienThiChan(pa,pn);
566     LaNguyenTo(pa,pn);
567     KhongLaNguyenTo(pa,pn);
568     getch();
569 }
570
571 //Ham nhap mang su dung con tro
572 void NhapMang(int* pa, int *pn)
573 {
574     int i;
575     printf("\nNhap vao so phan tu cua mang:");
576     scanf("%d",pn);
577
578     for(i = 0; i < *pn; i++) //lay gia tri cua vung nho ma con tro pn dang tro toi
579     {
580         printf("a[%d] = ",i);
581         scanf("%d",pa+i);    //luu gia tri tai dia chi thu i cua mang
582     }
583 }
584 //Ham xuat mang su dung con tro
585 void XuatMang(int *pa, int *pn)
586 {
587     int i;
588     for(i = 0; i < *pn; i++)
589     {
590         printf("%3d",*(pa+i)); //xuat gia tri tai vi tri i cua mang
591     }
592 }
593
594 //Ham kiem tra chan le
595 //Tra ve 0: chan
596 //Tra ve 1: le
597 int ChanLe(int a)
598 {
599     if(a % 2 == 0)
600         return 0;
601     return 1;
602 }
603 //Ham dem so luong phan tu co gia tri le va in len man hinh
604 void HienThiLe(int *pa, int *pn)
605 {
606     int i;
607     int le = 0; //dem so phan tu le trong mang
608     printf("\nSo le:");
609     for(i = 0; i < *pn; i++)
610         if(ChanLe(*(pa+i)) == 1)
611         {
612             le++;
613             printf("%3d",*(pa+i));
614         }
615     printf("---->So phan tu le trong mang la: %d",le);
616 }

```

```

617 //Ham dem so luong phan tu co gia tri chan va in len man hinh
618 void HienThiChan(int *pa, int *pn)
619 {
620     int i;
621     int chan = 0; //dem so phan tu le trong mang
622     printf("\nSo chan:");
623     for(i = 0; i < *pn; i++)
624         if(ChanLe(*(pa+i)) == 0)
625         {
626             chan++;
627             printf("%3d",*(pa+i));
628         }
629     printf("--->So phan tu chan trong mang la: %d",chan);
630 }
631 //Ham kiem tra so nguyen to
632 //Tra ve 0: neu ko phai la so nguyen to
633 //Tra ve 1: neu la so nguyen to
634 int KTNguyenTo(int n)
635 {
636     int i;
637     if(n == 1) //1 khong phai la so nguyen to
638         return 0;
639     for(i = 2; i <= n/2; i++)
640         if(n % i == 0) //Neu chia het
641             return 0;
642     return 1; //la so nguyen to
643 }
644 //Ham in cac so nguyen to trong mang
645 void LaNguyenTo(int *pa, int *pn)
646 {
647     int i;
648     printf("\nSo nguyen to trong mang la:");
649     for(i = 0; i < *pn; i++)
650         if(KTNguyenTo(*(pa+i)) == 1)
651             printf("%3d",*(pa+i));
652 }
653 //Ham in cac so khong phai so nguyen to trong mang
654 void KhongLaNguyenTo(int *pa, int *pn)
655 {
656     int i;
657     printf("\nSo khong phai nguyen to trong mang la:");
658     for(i = 0; i < *pn; i++)
659         if(KTNguyenTo(*(pa+i)) == 0)
660             printf("%3d",*(pa+i));
661 }
662 /*****
663 *****/
664 /*Bai 8:
665 Chuong trinh tinh tong binh phuong cua cac so am trong 1 mang cac so nguyen
666 */
667 #include <stdio.h>
668 #include <conio.h>
669 #include <math.h>
670 #include <stdlib.h>
671
672 //Khai bao nguyen mau ham

```

```

673 void NhapMang(int *pa, int *pn);
674 void XuatMang(int *pa, int *pn);
675 long TongAm(int *pa, int *pn);
676
677 void main()
678 {
679     int *pa; //Khai bao con tro pa chi den mang a
680     int *pn ; //Khai bao con tro pn chi den bien n
681     pa = (int*) calloc(100,sizeof(int)); //Khoi tao vung nho chua 100 bien int
682     pn = (int*) malloc(sizeof(int));      //Khoi tao vung nho chua 1 bien int
683     clrscr();
684     //Goi ham nhap mang:
685     NhapMang(pa,pn);
686     //Xuat mang da nhap
687     printf("\nMang da nhap la:");
688     XuatMang(pa,pn);
689     printf("\nTong binh phuong cua cac so am: %d",TongAm(pa,pn));
690     getch();
691 }
692
693 //Ham nhap mang su dung con tro
694 void NhapMang(int* pa, int *pn)
695 {
696     int i;
697     printf("\nNhap vao so phan tu cua mang:");
698     scanf("%d",pn);
699
700     for(i = 0; i < *pn; i++) //lay gia tri cua vung nho ma con tro pn dang tro toi
701     {
702         printf("a[%d] = ",i);
703         scanf("%d",pa+i);    //luu gia tri tai dia chi thu i cua mang
704     }
705 }
706 //Ham xuat mang su dung con tro
707 void XuatMang(int *pa, int *pn)
708 {
709     int i;
710     for(i = 0; i < *pn; i++)
711     {
712         printf("%3d",*(pa+i)); //xuat gia tri tai vi tri i cua mang
713     }
714 }
715
716 //Ham tinh tong binh phuong cac so am trong mang
717 long TongAm(int* pa, int *pn)
718 {
719     int i;
720     long tong = 0;    //Tong binh phuong cua cac so am
721     for(i = 0; i < *pn; i++)
722         if(*(pa+i) < 0)    //Neu no la so am
723         {
724             tong += pow(*(pa+i),2);
725         }
726     return tong;
727 }
728 /*****

```

```

729  *****/
730  /*Bai 9:
731  Chuong trinh thuc hien viec dao 1 mang 1 chieu
732  */
733  #include <stdio.h>
734  #include <conio.h>
735  #include <stdlib.h>
736
737  //Khai bao nguyen mau ham
738  void NhapMang(int *pa, int *pn);
739  void XuatMang(int *pa, int *pn);
740  void DaoMang(int *pa, int *pn);
741
742  void main()
743  {
744      int *pa; //Khai bao con tro pa chi den mang a
745      int *pn ; //Khai bao con tro pn chi den bien n
746      pa = (int*) calloc(100,sizeof(int)); //Khoi tao vung nho chua 100 bien int
747      pn = (int*) malloc(sizeof(int));      //Khoi tao vung nho chua 1 bien int
748      clrscr();
749      //Goi ham nhap mang:
750      NhapMang(pa,pn);
751      //Xuat mang da nhap
752      printf("\nMang da nhap la:");
753      XuatMang(pa,pn);
754      printf("\nMang sau khi dao la:");
755      DaoMang(pa,pn);
756      XuatMang(pa,pn);
757      getch();
758  }
759
760  //Ham nhap mang su dung con tro
761  void NhapMang(int* pa, int *pn)
762  {
763      int i;
764      printf("\nNhap vao so phan tu cua mang:");
765      scanf("%d",pn);
766
767      for(i = 0; i < *pn; i++) //lay gia tri cua vung nho ma con tro pn dang tro toi
768      {
769          printf("a[%d] = ",i);
770          scanf("%d",pa+i);    //luu gia tri tai dia chi thu i cua mang
771      }
772  }
773  //Ham xuat mang su dung con tro
774  void XuatMang(int *pa, int *pn)
775  {
776      int i;
777      for(i = 0; i < *pn; i++)
778      {
779          printf("%3d",*(pa+i)); //xuat gia tri tai vi tri i cua mang
780      }
781  }
782  //Ham hoan vi 2 so nguyen
783  void HoanVi(int *a, int *b)
784  {

```

```

785     int t;
786     t = *a; /*a: lay gia tri tai o nho ma con tro a dang tro toi
787     *a = *b; /*b: lay gia tri tai o nho ma con tro b dang tro toi
788     *b = t;
789 }
790 //Ham dao nguoc mang 1 chieu
791 void DaoMang(int *pa, int *pn)
792 {
793     int i;
794     for(i = 0; i < (*pn)/2; i++)
795         HoanVi((pa+i), (pa+(*pn)-1-i));
796 }
797 /*****
798 *****/
799 /*Bai 10:
800 Chuong trinh nhap vao 2 ma tran co n dong va m cot
801 ->Thuc hien phep cong hai ma tran da nhap
802 ->Xuat ma tran cong len man hinh
803 */
804 #include <stdio.h>
805 #include <conio.h>
806 #include <stdlib.h>
807
808 //Khai bao nguyen mau ham:
809 void NhapMaTran(int *pa, int *pn, int *pm);
810 void XuatMaTran(int *pa, int *pn, int *pm);
811 void CongMaTran(int *pa, int *pb, int *pc, int *pn, int *pm);
812
813
814 void main()
815 {
816     int *pa; //Khai bao con tro pa chi den ma tran a
817     int *pb; //Khai bao con tro pb chi den ma tran b
818     int *pc; //Khai bao con tro pc chi den ma tran c
819     int *pn; //Khai bao con tro pn chi den bien n(dong)
820     int *pm; //Khai bao con tro pm chi den bien m(cot)
821     pa = (int*) calloc(100*100, sizeof(int)); //Khoi tao vung nho chua 10000 bien int
822     pb = (int*) calloc(100*100, sizeof(int)); //Khoi tao vung nho chua 10000 bien int
823     pc = (int*) calloc(100*100, sizeof(int)); //Khoi tao vung nho chua 10000 bien int
824     pn = (int*) malloc(sizeof(int)); //Khoi tao vung nho chua 1 bien int
825     pm = (int*) malloc(sizeof(int)); //Khoi tao vung nho chua 1 bien int
826
827     printf("\nNhap vao so dong:");
828     scanf("%d", pn); //pn chinh la dia chi cua bien n
829     printf("\nNhap vao so cot:");
830     scanf("%d", pm); //pm chinh la dia chi cua bien m
831
832     //Nhap cac phan tu cho ma tran a
833     printf("\nNhap vao ma tran a:");
834     NhapMaTran(pa, pn, pm);
835     //Nhap cac phan tu cho ma tran b
836     printf("\nNhap vao ma tran b:");
837     NhapMaTran(pb, pn, pm);
838
839     printf("\nMa tran a da nhap la:");
840     XuatMaTran(pa, pn, pm);

```

```

841     printf("\nMa tran b da nhap la:");
842     XuatMaTran(pb,pn,pm);
843     CongMaTran(pa,pb,pc,pn,pm);
844     printf("\nMa tran cong la:");
845     XuatMaTran(pc,pn,pm);
846     getch();
847 }
848 //Ham nhap ma tran su dung con tro
849 void NhapMaTran(int *pa, int *pn,int *pm)
850 {
851     for(int i= 0; i < *pn; i++)
852         for(int j = 0; j < *pm; j++)
853             {
854                 printf("\nx[%d][%d]=",i,j);
855                 scanf("%d", (pa+i*(*pm)+j));
856             }
857
858 }
859 //Ham xuat ma tran su dung con tro
860 void XuatMaTran(int *pa, int *pn,int *pm)
861 {
862     int i,j;
863     for(i= 0; i < *pn; i++)
864     {
865         printf("\n");
866         for(j = 0; j < *pm; j++)
867             {
868                 printf("%3d",*(pa+i*(*pm)+j)) ;
869             }
870     }
871 }
872 //Ham cong hai ma tran voi nhau
873 void CongMaTran(int *pa,int *pb, int *pc, int *pn,int *pm)
874 {
875     int i,j;
876     for(i =0 ;i<*pn;i++)
877         for(j=0;j<*pm;j++)
878             {
879                 *(pc+i * (*pm) + j) = *(pa+i * (*pm) + j) +(*(pb+i * (*pm) + j));
880             }
881 }
882
883

```