```cpp
//Dylan Messerly, CS 2318-002, Assignment 2 Part 2
//10/27/2020

#include <iostream>
using namespace std;

int a1[12],
    a2[12],
    a3[12];
char einStr[]    = "Enter integer #";
char moStr[]     = "Max of ";
char ieStr[]     = " ints entered...";
char emiStr[]    = "Enter more ints? (n or N = no, others = yes) ";
char begA1Str[]  = "beginning a1: ";
char nn09A1Str[] = "a1 (noneg09): ";
char procA1Str[] = "processed a1: ";
char procA2Str[] = "          a2: ";
char procA3Str[] = "          a3: ";
char dacStr[]    = "Do another case? (n or N = no, others = yes) ";
char dlStr[]     = "===============================";
char byeStr[]    = "bye...";

int main()
{
                char reply;
                int used1,
                    used2,
                    used3,
                    target,
                    intHolder,
                    count,
                    iter,
                    *hopPtr1,
                    *hopPtr11,
                    *hopPtr2,
                    *hopPtr3,
                    *endPtr1,
                    *endPtr2,
                    *endPtr3;

                reply = 'y';
                //while (reply != 'n' && reply != 'N')
                goto WTest1;
begW1://         {
                    used1 = 0;
                    hopPtr1 = a1;
                    //while (reply != 'n' && reply != 'N')
                    goto WTest2;
begW2://             {
                        cout << einStr;
                        cout << (used1 + 1);
                        cout << ':' << ' ';
                        cin >> *hopPtr1;
                        ++used1;
                        ++hopPtr1;

                        //if (used1 < 12)
                        if( used1 >=12 ) goto else1;
begI1://                 {
                            cout << emiStr;
                            cin >> reply;

                            goto endI1;
//                       }
else1://                 else
//                       {
                            cout << moStr << 12 << ieStr << endl;
                            reply = 'n';
endI1://                 }

WTest2:          //(reply != 'n' && reply != 'N')
                 //if (reply != 'n' && reply != 'N') goto begW2;
                 if (reply == 'n') goto xitW2;
                 if (reply != 'N') goto begW2;
endW2://         }
xitW2:

                  cout << begA1Str;
                  //if (used1 > 0)
                  if( used1 <=0 ) goto endI2;
begI2://          {
                      hopPtr1 = a1;
                      endPtr1 = a1 + used1;
                      //do
```

```
85   begDW1://                    {
86                                    cout << *hopPtr1 << ' ' << ' ';
87                                    ++hopPtr1;
88
89   endDW1://                    }
90   DWTest1:                     //while (hopPtr1 < endPtr1);
91                                if (hopPtr1 < endPtr1) goto begDW1;
92   xitDW1:
93
94   endI2://                 }
95                        cout << endl;
96
97                        //if (used1 > 0)  //begI3
98                        if (used1 <= 0) goto endI3;
99   begI3://                 {
100                              //for (hopPtr1 = a1, endPtr1 = a1 + used1;
101                              //                  hopPtr1 < endPtr1;
102                              //                      ++hopPtr1)
103                              hopPtr1 = a1;
104                              endPtr1 = a1 + used1;
105                              goto FTest1;
106  begF1://                     {
107                                  target = *hopPtr1;
108
109                                  //if (target < 0 || target > 9)
110                                  if (target < 0) goto begI4;
111                                  if (target <= 9) goto endI4;
112  begI4://                         {
113                                      //for (hopPtr11 = hopPtr1 + 1;
114                                      //        hopPtr11 < endPtr1;
115                                      //            ++hopPtr11)
116                              hopPtr11 = hopPtr1 + 1;
117                              goto FTest2;
118  begF2://                         {
119                                       *(hopPtr11 - 1) = *hopPtr11;
120                                       ++hopPtr11; //update
121  FTest2:                          if (hopPtr11 < endPtr1) goto begF2;
122  endF2://                          }
123                                      --used1;
124                                      --endPtr1;
125                                      --hopPtr1;
126  endI4://                     } //I4
127
128                              ++hopPtr1; // update
129  FTest1:                      if (hopPtr1 < endPtr1) goto begF1;
130  endF1://                     }
131                              cout << nn09A1Str;
132                              //if (used1 > 0)
133                              if (used1 <= 0) goto endI5;
134  begI5://                     {
135                                  hopPtr1 = a1;
136                                  endPtr1 = a1 + used1;
137                                  //do
138  begDW2://                         {
139                                      cout << *hopPtr1 << ' ' << ' ';
140                                      ++hopPtr1;
141  endDW2://                         }
142  DWTest2:                          // while (hopPtr1 < endPtr1);
143                                    if(hopPtr1 < endPtr1) goto begDW2;
144  xitDW2:
145
146  endI5://                      }
147
148                              cout << endl;
149
150                              used2 = 0;
151                              used3 = 0;
152                              hopPtr1 = a1;
153                              hopPtr2 = a2;
154                              hopPtr3 = a3;
155                              endPtr1 = a1 + used1;
156                              //while (hopPtr1 < endPtr1)
157                              goto WTest3;
158  begW3://                     {
159                                  intHolder = *hopPtr1;
160                                  *hopPtr2 = intHolder;
161                                  ++used2;
162                                  ++hopPtr2;
163                                  *hopPtr3 = intHolder;
164                                  ++used3;
165                                  ++hopPtr3;
166                                  ++hopPtr1;
167  WTest3:                       if(hopPtr1 < endPtr1) goto begW3;
168  endW3://                     }
```

```
169
170                            iter = 0;
171
172                            //do
173    begDW3://              {
174                                ++iter;
175                                count = 0;
176                                //if (iter == 1)
177                                if(iter != 1) goto else6;
178    begI6://                  {
179                                    //for (hopPtr1 = a1, endPtr1 = a1 + used1;
180                                    //                    hopPtr1 < endPtr1;
181                                    //                          ++hopPtr1)
182                                    hopPtr1 = a1;
183                                    endPtr1 = a1 + used1;
184                                    goto FTest3;
185    begF3://                      {
186                                        target = *hopPtr1;
187                                        //if (target != 5)
188                                        if (target == 5)  goto else7;
189    begI7://                          {
190                                            ++count;
191                                            goto endI7;
192    //                                 }
193    else7://                          else
194    //                                {
195                                            //if (count != 0)
196                                            if (count == 0) goto endI8;
197    begI8://                          {
198                                                *(hopPtr1 - count) = *hopPtr1;
199    endI8://                          }
200
201    endI7://                      }
202                                    ++hopPtr1;
203    FTest3:                        if(hopPtr1 < endPtr1) goto begF3;
204    endF3://                      }
205                                    used1 -= count;
206                                    //if (used1 == 0)
207                                    if (used1 != 0) goto endI9;
208    begI9://                      {
209                                        hopPtr1 = a1;
210                                        *hopPtr1 = -99;
211                                        ++used1;
212    endI9://                      }
213                                goto endI6;
214    //                         }
215    else6://                   else
216    //                         {
217                                    //if (iter == 2)
218                                    if (iter != 2) goto else10;
219    begI10://                      {
220                                        //for (hopPtr2 = a2, endPtr2 = a2 + used2;
221                                        //                    hopPtr2 < endPtr2;
222                                        //                          ++hopPtr2)
223                                        hopPtr2 = a2;
224                                        endPtr2 = a2 + used2;
225                                        goto FTest4;
226    begF4://                          {
227                                            target = *hopPtr2;
228                                            //if (target > 4)    //I11
229                                            if (target <= 4) goto else11;
230    begI11://                          {
231                                                ++count;
232                                                goto endI11;
233    //                                 }
234    else11://                          else //else11
235    //                                {
236                                                //if (count != 0)   //I12
237                                                if (count == 0) goto endI12;
238    begI12://                          {
239                                                    *(hopPtr2 - count) = *hopPtr2;
240    endI12://                          }  //I12
241
242    endI11://                          }  //I11 end brack
243
244                                        ++hopPtr2; //update
245    FTest4:                            if(hopPtr2 < endPtr2) goto begF4;
246    endF4://                          }
247
248                                        used2 -= count;
249
250                                        //if (used2 == 0)  //I13
251                                        if (used2 != 0) goto endI13;
252    begI13://                          {
```

```
253
254                                        hopPtr2 = a2;
255                                        *hopPtr2 = -99;
256                                        ++used2;
257
258   endI13://                        }  //I13
259
260                                goto endI10;
261   //                         }
262   else10://                    else
263   //                         {
264                                  //for (hopPtr3 = a3, endPtr3 = a3 + used3;
265                                  //                    hopPtr3 < endPtr3;
266                                  //                           ++hopPtr3)
267                                  hopPtr3 = a3;
268                                  endPtr3 = a3 + used3;
269                                  goto FTest5;
270   begF5://                     {
271                                    target = *hopPtr3;
272
273                                    //if (target < 6)    //I14
274                                    if (target >=6) goto else14;
275   begI14://                      {
276                                      ++count;
277                                      goto endI14;
278   //                           }
279   else14://                      else
280   //                           {
281                                      //if (count != 0)     //I15
282                                      if (count == 0) goto endI15;
283   begI15://                        {
284                                        *(hopPtr3 - count) = *hopPtr3;
285   endI15://                        }  //endI15
286
287   endI14://                      } //endI14
288
289                                    ++hopPtr3; //update
290   FTest5:                        if (hopPtr3 < endPtr3) goto begF5;
291   endF5://                     }
292                                  used3 -= count;
293                                  //if (used3 == 0) //I16
294                                  if (used3 != 0) goto endI16;
295   begI16://                    {
296                                    hopPtr3 = a3;
297                                    *hopPtr3 = -99;
298                                    ++used3;
299   endI16://                    }
300
301   endI10://                  } //I10
302   endI6://                 } //I6
303   endDW3://              }
304   DWTest3:              //while (iter < 3);
305                         if (iter < 3) goto begDW3;
306   xitDW3:
307   endI3://            }
308
309                     cout << procA1Str;
310                     //if (used1 > 0)
311                     if (used1 <= 0) goto endI17;
312   begI17://           {
313                         hopPtr1 = a1;
314                         endPtr1 = a1 + used1;
315                         //do
316   begDW4://             {
317                           cout << *hopPtr1 << ' ' << ' ';
318                           ++hopPtr1;
319   endDW4://             }
320   DWTest4:              //while (hopPtr1 < endPtr1);
321                         if(hopPtr1 < endPtr1) goto begDW4;
322   xitDW4:
323   endI17://           }
324                     cout << endl;
325
326                     cout << procA2Str;
327                     //if (used2 > 0)
328                     if (used2 <= 0) goto endI18;
329   begI18://           {
330                         hopPtr2 = a2;
331                         endPtr2 = a2 + used2;
332                         //do
333   begDW5://             {
334                           cout << *hopPtr2 << ' ' << ' ';
335                           ++hopPtr2;
336   endDW5://             }
```

```
337    DWTest5://            while (hopPtr2 < endPtr2);
338                          if (hopPtr2 < endPtr2) goto begDW5;
339    xitDW5:
340
341    endI18://             }  //endI18
342
343                      cout << endl;
344                      cout << procA3Str;
345                      //if (used3 > 0)
346                      if (used3 <= 0) goto endI19;
347    begI19://             { //Beg I 19
348                          hopPtr3 = a3;
349                          endPtr3 = a3 + used3;
350                          //do
351    begDW6://             {
352                              cout << *hopPtr3 << ' ' << ' ';
353                              ++hopPtr3;
354    endDW6://             }
355    DWTest6://            while (hopPtr3 < endPtr3);
356                          if (hopPtr3 < endPtr3) goto begDW6;
357    xitDW6:
358    endI19://             } //I19
359
360                      cout << endl;
361
362                      cout << dacStr;
363                      cin >> reply;
364
365    WTest1:          //(reply != 'n' && reply != 'N')
366                     //if (reply != 'n' && reply != 'N') goto begW1;
367                     if (reply == 'n') goto xitW1;
368                     if (reply != 'N') goto begW1;
369
370    endW1://         }
371    xitW1:
372                     cout << dlStr << '\n';
373                     cout << byeStr << '\n';
374                     cout << dlStr << '\n';
375
376                     return 0;
377    }
378
```