```
#include <cstdio>
struct student
{
  int charm;
  int require;
} b[405], g[405];
struct edge
{
  int destination;
  edge *next;
} e[400 * 400];
int cnt;
edge *u[405];
int vis[405], match[405];
int res = 0;
bool Hungarian(int x)
{
  edge *p = u[x];
  while (p != NULL){
    if (vis[p->destination] == 0){
      vis[p->destination] = 1;
      if (match[p->destination] == 0 || Hungarian(match[p->destination])){
        match[p->destination] = x;
        return 1;
      }
    }
    p = p->next;
  }
  return 0;
}
int main()
{
  int n;
  scanf("%d", &n);
  for (int i = 1; i <= n; i++) scanf("%d", &b[i].charm);
  for (int i = 1; i <= n; i++) scanf("%d", &b[i].require);
  for (int i = 1; i <= n; i++) scanf("%d", &g[i].charm);
  for (int i = 1; i <= n; i++) scanf("%d", &g[i].require);
  for (int i = 1; i <= n; i++) u[i] = NULL;
  for (int i = 1; i <= n; i++){
    for (int j = 1; j <= n; j++){
      if (b[i].charm >= g[j].require && b[i].require <= g[j].charm){
        e[++cnt].destination = j, e[cnt].next = NULL;
        if (u[i] == NULL) u[i] = &e[cnt];
        else e[cnt].next = u[i],u[i] = &e[cnt];
      }
    }
  }
  for (int i = 1; i <= n; i++){
    for (int j = 1; j <= n; j++) vis[j] = 0;
    if (Hungarian(i)) res++;
  }
  printf("%d", res);
  return 0;
}
```