#include<stdio.h>

//#include<conio.h>

// A job has start time, finish time and profit.

struct activitiy

{

int start, finish;

};

typedef struct activitiy Activity;

// A utility function that is used for sorting activities according to finish time

void sort( Activity activities[ ], int n )

{

int i, j;

Activity t;

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

{

for( j=i+1 ; j<n ; j++ )

{

if( activities[i].finish > activities[j].finish )

{

t = activities[i];

activities[i] = activities[j];

activities[j] = t;

}

}

}

}

// Returns count of maximum set of activities that can be done by a single person, one at a time.

void printMaxActivities(Activity activities[ ], int n)

{

int i, j;

// Sort jobs according to finish time

sort( activities, n );

printf( "\n\n Following activities are selected \n\n" );

// The first activity always gets selected

i = 0;

printf ( " ( %d , %d ) ", activities[i].start , activities[i].finish );

// Consider rest of the activities

for ( j = 1; j < n; j++)

{

// If this activity has start time greater than or

// equal to the finish time of previously selected

// activity, then select it

if (activities[j].start >= activities[i].finish)

{

printf ( " ( %d , %d ) ", activities[j].start , activities[j].finish );

i = j;

}

}

}

// Driver program

void main()

{

Activity activities[] = { {10,13},

{9,14},

{7,11},

{12,16},

{20,25},

{1,50} };

int n = sizeof( activities )/sizeof( activities[0] );

// clrscr();

printMaxActivities( activities , n);

// getch();

}