Answer on Question #61018, Programming & Computer Science / C++

TCS is working on a new project called "TestVita". There are N modules in the project. Each module (i) has completion time denoted in number of hours (Hi) and may depend on other modules. If Module x depends on Module y then one needs to complete y before x.

As Project manager, you are asked to deliver the project as early as possible. Provide an estimation of amount of time required to complete the project.

Solution:

Let x_1, x_2, ... x_N – all modules, H_1, H_2, ... H_N – corresponding completion times, $F(x_i,x_j) = 1$ if x_i depends on module x_j, $F(x_i,x_j) = 0$ if x_j doesn't depend (directly) on module x_i.

Let x_k – the main module of the project. Our task is to find all modules on which this x_k depends (directly or indirectly through sequence of some other modules) and then find the sum of corresponding H_i.

Let's mark all modules by p_i, where $p_i = 1$ if that module is necessary for project, $p_i = 0$ if not.

Solution algorithm:

1. Let $p_k = 1$, all other $p_i = 0$.
2. Loop through all pairs x_i, x_j. If module x_i depends on module x_j ($F(x_i,x_j) = 1$) AND module x_i marked as 1 ($p_i = 1$) then change p_j mark to 1.
3. If in step 2 at least one mark p_i is changed, then go to step 2 and repeat it. If no p_i is changed then proceed to step 4.
4. Now all modules necessary for project is marked by 1 ($p_i = 1$ for them). Loop through all x_i and sum such H_i for which $p_j = 1$.

   In C++ this algorithm can be coded for example in such way:

```
P[k] = 1;
Changed = 1;
while (Changed)
{
        Changed = 0;
        for (i=0; i<N; i++)
              for (j=0;j<N;j++)
                    if ((F[i][j] == 1) && (P[i] == 1))
                    {
```

```
                              Changed=1;
                              P[j] = 1;
                    }
          }
          for (i=0, sum=0; i<N; i++)
                    if (P[i] == 1) sum += H[i];
          cout << "Time to complete: " << sum;
```