|  |  |
| --- | --- |
| //Duplicate Knapsack  loop(i,X) LOOP(w,wt[i],W+1) MAX(dp[w], dp[w-wt[i]]+val[i]);  return dp[W]; | //0-1 Knapsack  loop(i,X) loop(w,W+1){  if(wt[i]<=w) dp[i+1][w] = max(dp[i][w], dp[i][w-wt[i]]+val[i]);  else dp[i+1][w]=dp[i][w];}  return dp[X][W]; |
| //LCSubString  loop(i,X) loop(j,Y){  if(S1[i]==S2[j]) MAX(maxLen, (dp[i+1][j+1]=dp[i][j]+1));  else dp[i+1][j+1]=0;}  return maxLen; | //LCSubSequ  loop(i,X) loop(j,Y){  if(S1[i]==S2[j]) dp[i+1][j+1]=dp[i][j]+1;  else dp[i+1][j+1] = max(dp[i+1][j],dp[i][j+1]);}  return dp[X][Y]; |
| //PalinSubString  LOOP(len,2,X+1) loop(i,X-len+1){  int j=i+len-1;  if(in[i]==in[j] && dp[i+1][j-1]) dp[i][j]=1, begin=i, maxLen=len;}  return maxLen; | //PalinSubSequ  LOOP(len,2,X+1) loop(i,X-len+1){  int j=i+len-1;  if(in[i]==in[j]) dp[i][j]=dp[i+1][j-1]+2;  else dp[i][j]=max(dp[i][j-1],dp[i+1][j]);}  return dp[0][X-1]; |
| //Matrix Mul  LOOP(len,2,X) loop(i,X-len){  int j=i+len-1;  dp[i][j]=INT\_MAX;  LOOP(k,i,j){  int t = dp[i][k]+dp[k+1][j] + dim[i]\*dim[k+1]\*dim[j+1];  if(t<dp[i][j]) dp[i][j]=t, dpf[i][j]=k+1;}}  return dp[0][X-2]; | //LIS  loop(i,X) dp[i]=1;  loop(i,X) loop(j,i) if(in[i]>in[j]) MAX(dp[i],dp[j]+1);  loop(i,X) MAX(maxLIS,dp[i]);  return maxLIS; |
| //floydWarshell  loop(i,X) loop(j,X) loop(k,X) MIN(G[i][j], G[i][k]+G[k][j]); | |
| //Edit Distance  ZERO(dp); loop(i,X) dp[i][0]=i; loop(j,Y) dp[0][j]=j;  loop(i,X) loop(j,Y) dp[i+1][j+1]=min(dp[i][j]+(s1[i]!=s2[j]),min(dp[i+1][j]+1,dp[i][j+1]+1));  return dp[X][Y]; | |