Óynas	mic F	rogramm	ning	
				F.
	Sequence	Sequence Alig	Sequence Alignment	

service de l'integralité des contra	A DNA strand consists of a string of
	A DNA strand consists of a string of molecules called bases
\$1.00 ATACH in June gamente	movecures (alled pases
#BHOPMAN HOUSE BOTH HOUSE	4 Types of bases:
a development of the control of the	4 Types of bases: Adenine A
	- Cytosine C
-	- Guarine G
million different recovering an administrative	
the minimum grant that the manager	- Thypiene T
and the second s	
Allendrinatespania	

	Suppose we have 2 strings X&Y.
X	$'=\{\chi_1\chi_2\ldots\chi_m\}$
Y	$= \{Y, Y_2, Y_n\}$
Def	A matching is a set of ordered points with property that each items occurs at most once.
Will deliver the second section of the section of the second section of the second section of the second section of the section of the second section of the	
3500 STAN USAN BERKARAN SANSAY KEETAN AMARIN SANSAY KEETAN BARAN BARAN SANSAY KEETAN BARAN B	

Def. A matching is an alignment

if there are no crossing pairs.

For an alignment M between X&Y
1- We incur a "gap penalty" of 8 for each gap.
2- For each mismatch (of letters ps q) we incur a mismatch cost & pq
pg

Def. Similarity between strings X&Y is the minimum cost of an alignment Detween X & Y.

$$\chi = \{\chi_1, \ldots, \chi_m\}$$

 $Y = \{Y_1, \dots, Y_n\}$

Say Mis an opt. solution. either (Xm, Yn) &M or (Xm, Yn) &M

Define OPT (

In an optimal alignment M. at	
In an optimal alignment M, at least one of the following is true:	stoc
1)- (Xm, Yn) = M -> OPT(M, n) = OPT(M-1	, 11-
2) - Xm is not matched -> OPT(m,n) =	production of the state of the
3) - In is not metched -> OPT(m,n) =	
OPT(m, n-1)+8	erent.
	mid.
	-
	-
	26
	ari
	ar











Alphanometric Control of Control	Matrix	Chain	Mul	tiplicates
				quano
	DAAATOO SIIAANA SIIN SIIN SIIN SIIN SIIN SIIN S			
			Account to the second s	
	Management of the second secon			







		December 1 of the Control of State of S
Shortest 1	Path Prob	be.
Dyname	Rogram	uming
		nes ann ag mar aire aidh a caith ag ainte dhaille aidh a Ba' ag air an aidh an ann an ann an ann an ann an ann an a
		Backado este Cido propriendados partidos está presidente de construir
		BOOK BOOK TO THE WITTER STORE THE STORE ST







Bellman-Ford Alg.
Shortest-path (G, s, t)
n = no. of nodes in G
define M[o,t]=0, M[o,v]=0
- for i=1 to n-1
for VEV in any order
M[i,v] = Min (M[i-1,v],
min (M[i-1, w] + Cvw))
end for (WE Adj(v)
end for









Discussion 7

- 1. When their respective sport is not in season, USC's student-athletes are very involved in their community, helping people and spreading goodwill for the school. Unfortunately, NCAA regulations limit each student-athlete to at most one community service project per semester, so the athletic department is not always able to help every deserving charity. For the upcoming semester, we have S student-athletes who want to volunteer their time, and B buses to help get them between campus and the location of their volunteering. There are F projects under consideration; project i requires s_i student-athletes and b_i buses to accomplish, and will generate g_i > 0 units of goodwill for the university. Our goal is to maximize the goodwill generated for the university subject to these constraints. Note that each project must be undertaken entirely or not done at all -- we cannot choose, for example, to do half of project i to get half of g_i goodwill.
- 2. Suppose you are organizing a company party. The corporation has a hierarchical ranking structure; that is, the CEO is the root node of the hierarchy tree, and the CEO's immediate subordinates are the children of the root node, and so on in this fashion. To keep the party fun for all involved, you will not invite any employee whose immediate superior is invited. Each employee j has a value v_j (a positive integer), representing how enjoyable their presence would be at the party. Our goal is to determine which employees to invite, subject to these constraints, to maximize the total value of invitees.
- **3.** You are given a set of n types of rectangular 3-D boxes, where the ith box has height h(i), width w(i) and depth d(i) (all real numbers). You want to create a stack of boxes which is as tall as possible, but you can only stack a box on top of another box if the dimensions of the 2-D base of the lower box are each strictly larger than those of the 2-D base of the higher box. Of course, you can rotate a box so that any side functions as its base. It is also allowable to use multiple instances of the same type of box.











