

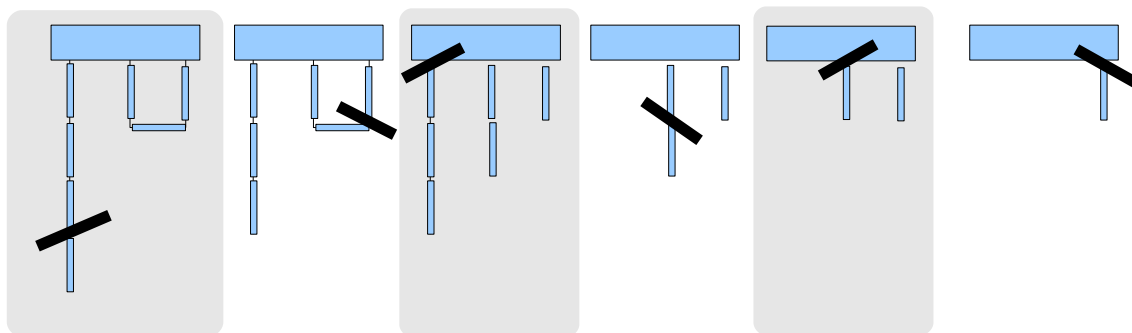
ซ้อมดาบ

พนมยงค์เป็นนักดาบที่ขยันและเอาจริงเอาจังมาก แม้กระทั่งเวลาฝึก เขายังต้องหาความท้าทายให้กับตัวเอง เขาฝึกหัดฟันดาบ โดยใช้ท่อนไม้ความยาวเท่า ๆ กัน หลาย ๆ ท่อนมาร้อยต่อกันเป็นสาย จากนั้นนำไปแขวนไว้กับกิ่งไม้ โดยเขาแขวนได้สองแบบคือ แขวนปลายด้านเดียว กับแขวนที่ปลายสายสองข้าง (ดูรูปด้านข้าง)

เขาสร้างเกมฟันดาบขึ้นเล่นกับเพื่อน ๆ ผู้เล่นแต่ละคนจะสามารถฟันเชือกให้ขาดได้ เมื่อเชือกขาด ท่อนไม้ด้านล่างก็จะร่วงลงสู่พื้น ในกรณีที่สายท่อนไม้เป็นการแขวนสองข้าง สายดังกล่าวจะขาดออกกลายเป็นสายท่อนไม้แบบแขวนด้านเดียวสองเส้น

ยกตัวอย่างเช่น ถ้าฟันสายไม้ด้านซ้ายจากที่ยาว 4 ท่อน อาจฟันให้เหลือ 0, 1, 2, หรือ 3 ท่อนได้ ส่วนสายไม้ด้านขวา เขาสามารถฟันให้กลายเป็นสายท่อนไม้ความยาว 1 กับสองท่อน หรือ ถ้าฟันที่เชือกปลายสุดสายท่อนไม้ดังกล่าว ก็จะทำให้ขาดเป็นสายท่อนไม้แบบแขวนปลายด้านเดียวที่มีท่อนไม้ 3 ท่อนได้

เป้าหมายของเกมนี้คือการฟันเชือกให้ท่อนไม้ร่วงลงสู่พื้นจนหมด ใครฟันได้ก่อนเป็นผู้ชนะ ตัวอย่างของการเล่นดังกล่าวแสดงด้านล่าง



พนมยงค์ฟัน

เพื่อนฟัน

พนมยงค์ฟัน

เพื่อนฟัน

พนมยงค์ฟัน

เพื่อนฟัน

เนื่องจากเพื่อนของพนมยงค์เป็นคนที่ฟันแทงไม้ร่วงหล่น เขาจึงเป็นผู้ชนะในการแข่งขันนี้

พนมยงค์เกิดความเศร้าเสียใจมากที่เล่นเกมนี้แพ้ จึงขอร้องให้คุณเขียนโปรแกรมเพื่อเล่นเกมนี้ให้กับเขา โดยในการเริ่มเล่นเกม

พนมยงค์จะเป็นคนเริ่มต้นก่อนเสมอ

การติดต่อกับไลบรารี

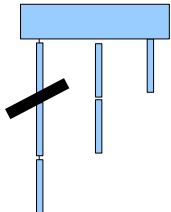
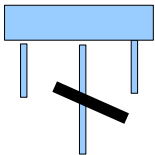
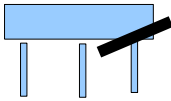
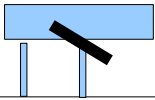
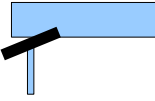
คุณจะได้รับไลบรารีในแฟ้ม swordlib.h และ swordlib.cpp สำหรับเขียนโปรแกรม โดยไลบรารีจะมีคำสั่งดังนี้

- **void sword_init(int* scount, int* slen, int* dcount, int* dlen)**
คืนจำนวนสายแท่งไม้แบบแขวนข้างเดียว (ในตำแหน่งที่ชี้โดย scount) และความยาวของสายแต่ละสายในอาร์เรย์ที่ชี้เริ่มต้นด้วย slen, และจำนวนสายแท่งไม้แบบแขวนสองข้าง (ในตำแหน่งที่ชี้โดย dcount) และความยาวของสายแต่ละสายในอาร์เรย์ที่ชี้ด้วย dlen จำนวนสายแท่งไม้รวมทั้งหมดไม่เกิน 100 เส้น จำนวนแท่งไม้ในแต่ละสายมีไม่เกิน 100 แท่ง
- **void sword_cut(int type, int orglen, int newlen)**
สั่งตัดสายแท่งไม้ ถ้า type = 1 จะตัดสายที่แขวนปลายด้านเดียว ถ้า type = 0 จะตัดสายที่แขวนปลายสองข้าง จะตัดสายที่มีความยาวเท่ากับ orglen จะตัดให้ส่วนด้านหนึ่งที่แขวนกับกิ่งไม้มีความยาว newlen ($0 \leq \text{newlen} < \text{orglen}$)
- **void sword_watch(int* type, int* orglen, int* newlen)**
อ่านข้อมูลการฟันเชือกของคู่แข่ง ในลักษณะเดียวกับที่ส่งในฟังก์ชัน cut

ไลบรารีตัวอย่างที่ให้ไป จะอ่านข้อมูลของสายแท่งไม้จาก standard input โดยจะรับข้อมูล 4 บรรทัด บรรทัดแรกและบรรทัดที่ 3 ระบุค่าของ scount และ dcount, บรรทัดที่ 2 และ 4 ระบุรายการความยาวของสายแท่งไม้

นอกจากนี้เมื่อสั่ง sword_watch จะรับค่าจาก standard input (โดยสั่ง scanf(“%d %d %d”,type,ordlen,newlen);) แต่ไลบรารีที่ใช้ในการตรวจจริง จะเล่นเป็นคู่ต่อสู้ที่จะเอาชนะคุณถ้าเป็นไปได้ ไลบรารีจะจบการทำงานของโปรแกรมของคุณเอง เมื่อคุณชนะ หรือไม่แพ้

ตัวอย่างการติดต่อ

คำสั่ง	ผล	คำอธิบาย
int scount, dcount; int slen[100], dlen[100]; sword_init(&scount, slen, &dcount, dlen);	scount = 1, slen = {4}, dcount = 1, dlen = {3}	
sword_cut(1,4,3);		
int t,u,v; sword_watch(&t,&u,&v);	t = 0, u = 3, v = 2	สามารถให้ v = 1 ก็ได้ผลแบบเดียวกัน
sword_cut(1,3,1);		
sword_watch(&t,&u,&v);	t = 1 u = 2 v = 1	
sword_cut(1,1,0);		
sword_watch(&t,&u,&v);	t = 1 u = 1 v = 0	
sword_cut(1,1,0);		 ชนะ

การให้คะแนน สำหรับทุก ๆ ข้อมูลชุดทดสอบ สามารถมีวิธีการฟันเชือกที่ทำให้คุณชนะได้เสมอ คุณจะได้คะแนนเต็มสำหรับชุดทดสอบถ้าคุณชนะได้ และจะได้ 0 ถ้าแพ้ หรือสั่งงานไลบรารีผิดพลาด

ข้อจำกัด โปรแกรมของคุณ เมื่อทำงานร่วมกับไลบรารีสำหรับเล่นเกมแล้ว จะต้องใช้เวลาการทำงานไม่เกิน 1 วินาที (ไลบรารีสำหรับเล่นเกมทำงานเร็วมาก) และหน่วยความจำไม่เกิน 128 MB