

Quiz-1

Algorithm Design - Spring 2026

Instructor: Dr. Amin Eskandari

Head Teaching Assistants: Hamid Namjoo, Amir Hossein Hemati

Teaching Assistants: Ali Nikvan, Viyana Hasanbeigi, Nima Soltani, Reza Rezaie, Maryam

Meidanshahi , Babak Janfaza , Fatemeh Dehgan , Elham Mosavi

سوال ۱ :

یک شرکت مخابراتی تصمیم دارد یک پروتکل جدید برای فشرده‌سازی، انتقال و بازیابی داده طراحی کند شما به عنوان تحلیل‌گر الگوریتم، باید برای بخش مهمی از این پروتکل تصمیم‌گیری کنید.

اطلاعات اولیه:

(۱) نوع داده‌ها

داده‌هایی که باید ارسال شوند، شامل کاراکترهایی با فراوانی متفاوت هستند فراوانی برخی از کاراکترها مشخص است اما تعدادی از آن‌ها ممکن است در طول انتقال حذف یا خراب شوند.

(۲) هزینه انتقال

برای ارسال هر بیت، هزینه انتقال متناسب با وزن یک بسته است شرکت تنها می‌تواند بسته‌هایی را ارسال کند که مجموع وزن آن‌ها از یک مقدار مشخص (W) بیشتر نشود هر کاراکتر پس از کدگذاری با هافمن دارای تعداد بیت متفاوت است، پس هزینه ارسال کاراکترها متفاوت خواهد بود

این محدودیت دقیقاً مشابه کوله‌پشتی کسری است: شما می‌توانید بخشی از داده‌ها را رمزگذاری و ارسال کنید و بخشی دیگر را کنار بگذارید.

(۳) نیاز به بازیابی کامل برخی پیام‌ها

در حین انتقال، برخی دنباله‌ها باید قابل دیکدینگ بدون ابهام باشند

اما یک شرط اضافه داریم: چون داده‌ها در چند بسته ارسال می‌شوند، اگر انتهای یک بسته با ابتدای بسته بعدی ترکیب شود، نباید یک رشته مبهم ایجاد شود. یعنی باید یک کد هافمن با خاصیت Instantaneous + Prefix-free + Packet-Safe بسازید.

(۴) پرداخت پول

برای ارسال هر کاراکتر، شرکت باید هزینه‌ای پرداخت کند. اما پرداخت‌ها فقط در واحدهایی از یک پول خاص انجام می‌شود (مثلاً سکه‌های ۱، ۳، ۴ واحدی)

شرکت می‌خواهد مجموع پرداخت‌ها حداقل باشد و شما باید تعیین کنید آیا می‌توان از طریق خرد کردن پول و انتخاب مناسب کاراکترهایی که ارسال می‌کنید، این هزینه را بهینه کرد.

شرح مسئله

برای مجموعه کاراکترهای زیر، فراوانی، وزن ارسال و هزینه پرداختی برای ارسال یک کاراکتر به صورت زیر است:

کاراکتر A: فراوانی = ۵۰، وزن = ۴، هزینه = ۵

کاراکتر B: فراوانی = ۴۰، وزن = ۳، هزینه = ۳

کاراکتر C: فراوانی = ۳۰، وزن = ۵، هزینه = ۴

کاراکتر D: فراوانی = ۲۰، وزن = ۲، هزینه = ۲

کاراکتر E: فراوانی = ۱۰، وزن = ۱، هزینه = ۱

حداکثر وزن قابل ارسال: $W = 180$

پرداخت فقط با سکه‌های {۱، ۳، ۴} انجام می‌شود

وظیفه شما

(۱) یک کد هافمن طراحی کنید که هم Prefix-free باشد و هم Packet-Safe

یعنی:

الف) هیچ کدی پیشوند دیگری نباشد

ب) هیچ کدی در انتهای بسته اول و ابتدای بسته بعدی با هم ترکیب نشده و یک کد معتبر دیگر نسازند

(۲) با استفاده از ساختار کد خود، تعیین کنید از هر کاراکتر چه مقدار باید ارسال شود تا:

محدودیت وزن کوله‌پشتی کسری رعایت شود

ارزش اطلاعاتی (مجموع فراوانی $\times$ طول هافمن) بیشینه شود

(۳) سپس تعیین کنید آیا هزینه پرداختی برای ارسال این مجموعه کاراکتر را می‌توان با سکه‌های

$\{1, 3, 4\}$ بهینه پرداخت کرد؟

و حداقل هزینه ممکن را بیابید

(۴) اثبات کنید که انتخاب شما از نظر:

هافمن

کوله‌پشتی کسری

دیکدینگ بدون ابهام

خرد کردن پول

بهینه است

سوال ۲ :

دانشگاه قصد دارد یک سیستم رمزگذاری و تحلیل داده طراحی کند که مراحل آن به صورت زیر انجام می شود شما باید خروجی نهایی این فرایند را محاسبه کنید

مرحله ۱ ساخت دنباله اولیه

یک رشته S با طول n تعریف می کنیم که در آن:

کاراکتر i ام برابر است با باقی مانده عدد $F(i)$ (اعداد فیبوناچی) تقسیم بر ϵ

سپس این باقی مانده را به حرفی از مجموعه $\{A,B,C,D\}$ نگاشت می کنیم:

$$A \rightarrow 0$$

$$B \rightarrow 1$$

$$C \rightarrow 2$$

$$D \rightarrow 3$$

برای $n = 12$ ، رشته S را بسازید

مرحله ۲ ساخت درخت هافمن

با توجه به فراوانی حروف در رشته S ، درخت هافمن را بسازید و کد هر حرف را تعیین کنید

مرحله ۳ رمزگذاری

رشته S را با استفاده از کدهای هافمن مرحله قبل رمز کنید و به رشته دودویی H برسید

مرحله ۴ دیکدینگ ناقص

به دلیل نقص شبکه، ۲۰ درصد از بیت های رشته H حذف شده اند (بدون ترتیب مشخص)

برای بازسازی، یک ماتریس دودویی M با ابعاد $k \times \epsilon$ تشکیل می دهیم که k برابر تعداد

بیت های باقی مانده است

بیت های H را از بالا به پایین و از چپ به راست در ماتریس قرار دهید (ردیف ها را پر

کنید)

حال دو وظیفه دارید:

۱- طول بلندترین زیردنباله صعودی (LIS) از ستون‌های ماتریس M (هر ستون یک عدد دودویی ۴بیتی است) را بیابید

۲- طول بلندترین زیررشته مشترک (LCS) بین ردیف اول و ردیف آخر ماتریس را بیابید

مرحله ۵ خروجی نهایی

خروجی نهایی سیستم برابر است با:

$$\text{Output} = \text{LIS_length} \times \text{LCS_length}$$

مقدار این خروجی را دقیق محاسبه کنید

سوال ۳ :

شرکت شما آرایه ای در اختیار دارد که در آن هر عدد نشان می‌دهد سود یا زیان فروش روزانه شبکه های نفت نسبت به روز قبل چقدر تغییر کرده است (ممکن است منفی باشد).

۱. روش خود را توضیح دهید که چگونه می‌توان با یک بار پیمایش آرایه، بازه‌ای از روزهای پشت سرهم را پیدا کرد که بیشترین مقدار حاصل از مجموع عناصر آن بازه را داشته باشد.

در توضیح خود مشخص کنید که مجموع جاری (current sum) چگونه تغییر می‌کند و در چه حالتی مقدار آن صفر می‌شود.

۲. اکنون فرض کنید می‌خواهید سه بازه جدا از هم انتخاب کنید به گونه‌ای که جمع سه مقدار حاصل از آن‌ها بیشینه شود.

توضیح دهید چگونه می‌توان از نتایج پیمایش‌های قبلی (مثل مقادیر به دست آمده از مجموع بیشینه در بخش‌های مختلف آرایه) برای ساختن این انتخاب استفاده کرد.

سوال ۴ :

یک قنادی قصد دارد برای دو جشن مختلف، از یک لیست مشترک شیرینی‌ها انتخاب کند.

هر شیرینی دو ویژگی دارد:

- ارزش (Value): میزان رضایت مشتری یا سود قنادی.
 - وزن (Weight): فضایی که در بسته مخصوص جشن اشغال می‌کند.
- بسته جشن اول یک ظرفیت وزنی W دارد، هر شیرینی را یا به طور کامل انتخاب می‌کنیم یا اصلاً نمی‌گیریم (امکان نصف کردن یا خرد کردن برای این جشن وجود ندارد).

داده‌ها (برای جشن اول):

نام شیرینی	ارزش (Value)	وزن (Weight)
(A) شیرینی گردویی	10	3
(B) شیرینی پسته‌ای	12	4
(C) شیرینی میوه‌ای	18	6
(D) شیرینی کاکائویی	22	7

ظرفیت بسته جشن اول:

$$W = 11$$

هدف: مجموع ارزش شیرینی های انتخاب شده بیشینه شود، طوری که مجموع وزن آنها از ۱۱ بیشتر نشود.

(الف)

۱. توضیح دهید چرا در این صورت مسئله نمی‌توانیم از ایده تقسیم‌کردن شیرینی‌ها مثل کوله پشتی کسری استفاده کنیم.

به زبان خودتان بگویید چه تفاوت مفهومی بین کوله‌پشتی کسری (fractional) و این حالت $1/0$ وجود دارد، اگر هر دو در ظاهر وزن و ارزش دارند.

۲. فرض کنید به اشتباه تصمیم بگیریم همین مسئله را با یک روش حریصانه‌ی ساده (Greedy) بر اساس نسبت ارزش به وزن حل کنیم.

بدون محاسبه دقیق همه حالت‌ها، توضیح دهید چرا این رویکرد در مسئله $1/0$ لزوماً جواب بهینه نمی‌دهد.

(ب)

ما می‌خواهیم برای این مسئله یک رابطه بازگشتی و سپس یک جدول DP در ذهن مان تجسم کنیم اما نیازی به نوشتن کد نیست.

۱. یکی از حالت‌های متداول این است که حالت را به صورت

$DP[i][w]$ تعریف کنیم به این معنا که:

«بیشترین ارزش قابل دستیابی با استفاده از شیرینی‌های ۱ تا i و با ظرفیت دقیقاً w چیست؟»

به زبان خودتان توضیح دهید این تعریف چه اطلاعاتی را نگه می‌دارد و چرا برای این مسئله مناسب است.

بر اساس همین تعریف، به صورت مفهومی بنویسید اگر در حال تصمیم‌گیری برای شیرینی i ام هستیم، چه دو انتخاب اصلی داریم و این چگونه به یک رابطه

بازگشتی منجر می‌شود (نیازی به نوشتن فرمول ریاضی دقیق نیست، توضیح کلامی کافی است).

۲. برای داده‌های جدول بالا و ظرفیت $W = 11$ ، با استفاده از منطق رابطه بازگشتی، تعیین کنید کدام ترکیب از شیرینی‌ها را برمی‌گزینید تا مجموع ارزش بیشینه شود.

نام شیرینی‌ها و مجموع ارزش نهایی را بنویسید و به‌طور کوتاه توضیح دهید بر اساس چه مقایسه‌هایی به این نتیجه رسیدید.

سوال ۵:

یک تیم تحقیقاتی در حال توسعه یک ربات تحویل‌دهنده هوشمند است که پیام‌های صوتی را فشرده و به مراکز مختلف می‌فرستد .

ربات پیام‌ها را با کدگذاری هافمن فشرده می‌کند .

برخی پیام‌ها باید سریع ارسال شوند اما ظرفیت ارسال ربات محدود است (مشابه کوله‌پشتی که وزن کل بسته‌ها نباید بیش از یک مقدار W باشد).

کدهای هافمن طراحی شده باید هم پیش‌وند رایگان (prefix-free) باشند تا صدای پیام‌ها به درستی دیکد شود .

محققان می‌خواهند از طریق تحلیل دنباله‌های پیام‌های مختلف، طول بلندترین دنباله مشترک (LCS) صوتی را بیابند تا کیفیت فشرده‌سازی را ارزیابی کنند .

-بودجه تیم تحقیقاتی محدود است، بنابراین هزینه ارسال پیام‌ها باید با استفاده از واحدهای پرداخت جدید (سکه‌های ۲، ۵ و ۷ واحدی) به حداقل برسد.

کار شما :

۱. داده‌های صوتی را به کمک کد هافمن به کدهای پیش‌وند رایگان تبدیل کنید (فرض کنید طول کد هر کاراکتر داده شده است) .
۲. با الگوریتم کوله‌پشتی صفر-یک، تعیین کنید کدام پیام‌ها و چه تعدادی باید ارسال شوند تا جمع وزن بسته‌ها از W تجاوز نکند و ارزش اطلاعاتی (درصد دقت صوتی) بیشینه شود .
۳. طول LCS دنباله دو پیام داده شده را بیابید .
۴. با توجه به سکه‌های موجود (۲، ۵، ۷ واحدی) حداقل هزینه پرداخت را تعیین و اثبات کنید که بهینه است.

سوال ۶ :

- در یک شبکه فضایی بین‌سیاره‌ای، اطلاعات حسگرهای مختلف به زمین منتقل می‌شود .
- هر داده با طول کد متفاوت پس از فشرده‌سازی هافمن ارسال می‌شود .
- به دلیل محدودیت انرژی، تنها مقدار مشخصی (W) انرژی صرف انتقال اطلاعات می‌شود (مدل کوله‌پشتی کسری) .
- ترتیب ارسال داده‌ها اهمیت دارد؛ به طوری که طول بلندترین دنباله صعودی (LIS) روی طول کدهای انتخابی باید حداکثر شود تا نرخ خطا در انتقال حداقل شود .
- هزینه انتقال هر بیت با واحدهای پرداخت نامتقارن (سکه‌های ۱، ۴ و ۶ واحدی) باید به شکلی خرد شود که کمترین هزینه کل پرداخت شود .

باید تضمین شود که کدهای هافمن قابل دیکدینگ بدون ابهام باشند و هنگام اتصال بسته‌ها مشکلی پیش نیاید.

کار شما :

۱. با توجه به فراوانی داده‌ها، طول کدهای هافمن را تعیین کنید .
۲. با مدل کوله‌پشتی کسری، تعداد هر داده برای ارسال را مشخص کنید به طوری که مجموع وزن از W بیشتر نشود و ارزش انتقال بیشینه باشد .
۳. بیشینه طول دنباله LIS روی توالی طول کدهای انتخاب شده را بیابید .
۴. کمینه هزینه پرداخت قابل خرد کردن با سکه‌های ۱، ۴ و ۶ واحد را محاسبه و اثبات بهینه بودن آن را انجام دهید .
۵. توضیح دهید چگونه می‌توانید تضمین کنید کدهای هافمن طی روند ارسال بدون ابهام قابل دیکد شدن هستند (مبحث پیش‌وند بودن + دیکدینگ لحظه‌ای).

سوال ۷:

در شبکه‌کد الگوریتم Huffman ، در هر مرحله دو گره کم‌وزن انتخاب می‌شوند:

Huffman(C):

Q = priority – queue of characters in C by frequency

while |Q| > 1:

x = Extract – Min(Q)

y = Extract – Min(Q)

create node z with weight = x.weight + y.weight

insert z into Q

return Q

الف) چرا این الگوریتم باید از min-priority queue استفاده کند؟

اگر از صف معمولی استفاده کنیم چه مشکلی پیش می‌آید؟

ب) پیچیدگی زمانی ساخت کُد هافمن با استفاده از priority-queue چیست؟

(فقط اسم مرتبه کافی است: $O(...)$)

سوال ۸ :

دانشکده در حال طراحی یک ماژول است که ابتدا روی داده‌ی حسگر «بیشترین مجموع زیرآرایه‌ی متوالی» را محاسبه کرده و سپس یک مرحله‌ی فشرده‌سازی ساده انجام می‌دهد.

آرایه‌ی حسگر:

$$A = [4, -2, 1, 3, -5, 4]$$

الف)

با نوشتن شبه‌کد الگوریتم Kadane (نسخه‌ی استاندارد)، مشخص کنید بیشترین مجموع زیرآرایه‌ی متوالی این آرایه چقدر است.

(خروجی نهایی + اجرای دستی روی آرایه لازم است.)

ب)

شبه‌کدی بنویسید که بررسی کند آیا در یک کُد هافمن، کُد یک نماد می‌تواند پیشوند کُد نماد دیگر باشد یا نه.

(فرض کنید ورودی یک لیست از رشته‌های باینری است؛ خروجی True/False)