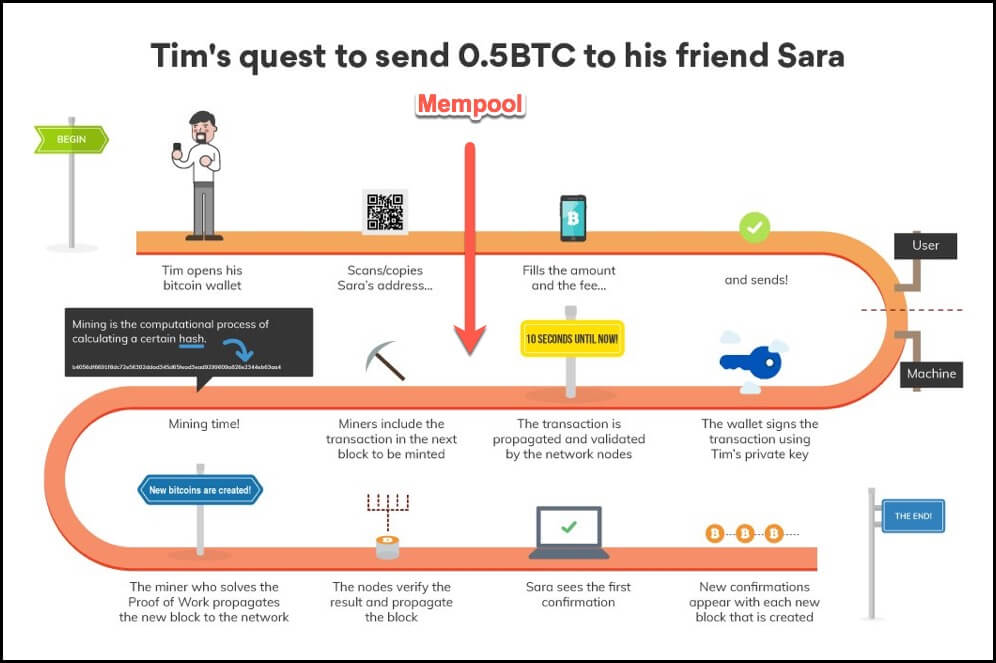
**העברות**

חלק העוסק בהעברות הכספים עצמם הוא החלק המורכב והגדול ביותר לכל הקשור לביטקוין. העברות הכספים חייבות להיות מדויקות, ייחודיות, ללא שגיאות ככל האפשר ומספיק מצומצמות כדי שכל העברה כזאת תוכל להגיע לכל רשת הביטקוין במהירות ובקלות.

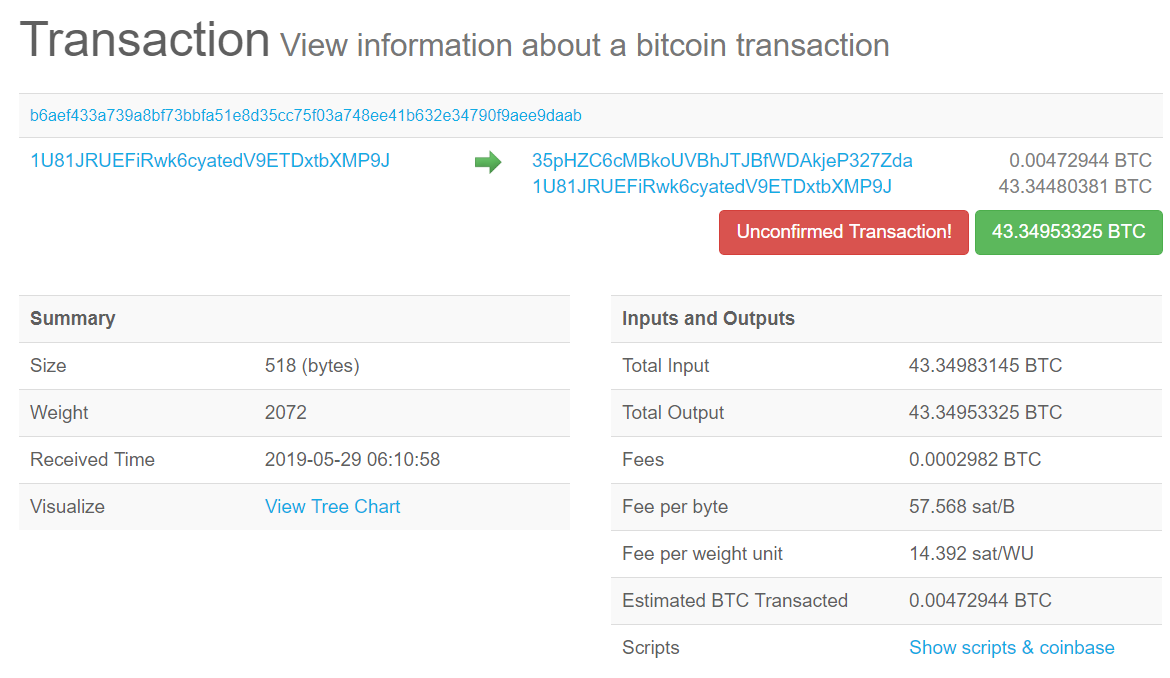


ראשית, נבין שבכל העברה – מועבר **כל** סכום הביטקוין שנמצא בכתובת (לא משנה בסכום ההעברה) אלא שרק סכום העברה מועבר לכתובת הרצויה ושאר הסכום מועבר אל הכתובת הציבורית של המשלם חזרה.

כדי להבין את הרציונל טוב יותר, נחשוב על דוגמה –

אדם רוצה לקנות תפוח שעולה 2 שקלים אבל יש לו רק שטר של 20, ישלם למוכר את כל ה20, המוכר יקבל רק 2 שקלים מתוכם ויחזיר 18 למשלם.

נתבונן בנתוני העברה שיש במוניטור אונליין:



נראה שבכתובות הנמענים יש את כתובת השולח גם היא, ואליה מועבר שאר הסכום שנמצא בכתובת.

נוכל לראות שאפשר להסיק חלק מהנתונים מבלי הצגה מפורשת שלהם -

למשל, מתוך הסכום הכולל שיש בכתובת, נוכל לחסר את הסכום הנשלח בפועל (המופיע בירוק) ולקבל את העמלה (Fees) ששולמה עבור העסקה.

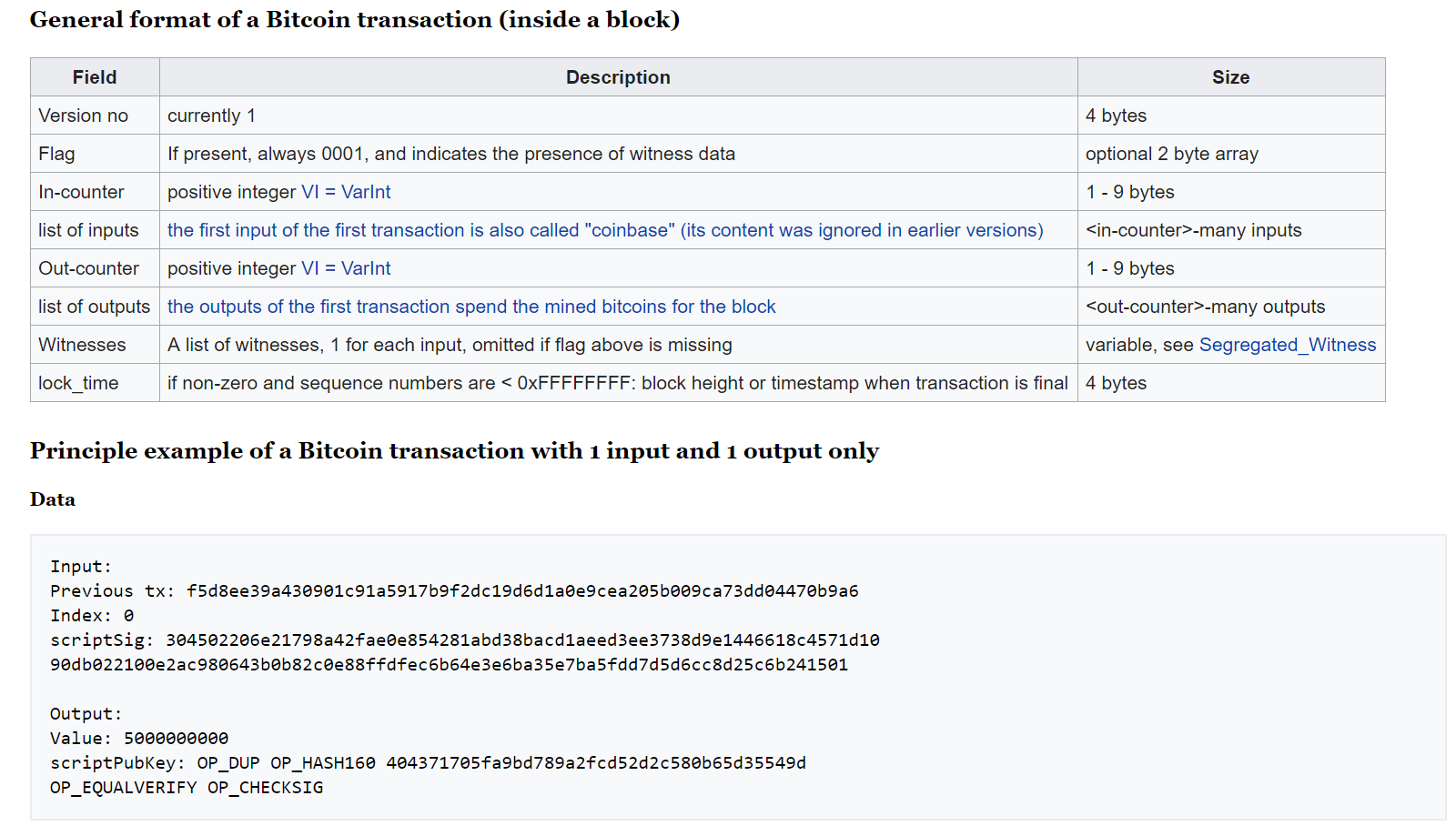
את הסכום הנשלח בפועל עצמו נוכל גם כן להסיק מתוך הכתובות המופיעות בנמענים ולסכום אותם. נוכל כבר לראות שיש שדות מיותרים. ברשת תקשורת עמוסה כמו ביטקוין נרצה לצמצם ככל שניתן את תכולת ההעברות – לכן:

היה אפשר לחשוב שכל הנתונים המוצגים מעלה הם חלק מנתוני ההעברה ומופיעים בכל מסר של העברה מחדש. למעשה, חלק מהנתונים המוצגים אינם חלק מנתוני ההעברות, ומוצגים ע"י תוכנות לקוח שמסיקות את הנתונים בעיבוד שיכבה גבוהה (high level).

נתונים כמו כתובת המקור וכתובת הנמען, הסכום המועבר ע"י המשלם אינם מוצגים בכלל ע"י הביטים המועברים בפועל.

בביטים, אין מטבעות, אין "קבלה", אין מאזן כספים ואין כתובות(!), כל הדברים האלה מועבדים מתוך ההעברה ברמה גבוהה יותר של ארנק וממשק משתמש כדי להציג את הנתונים באופן מובן יותר ויעיל יותר לשימוש.

נתונים המועברים בפועל להלן:



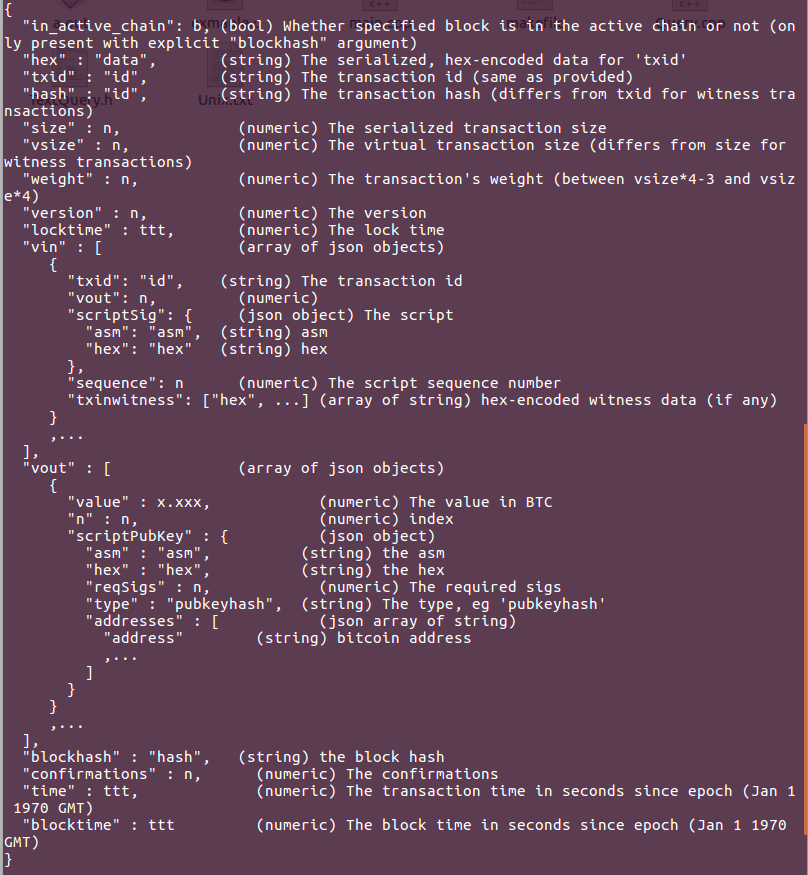
<https://bitcoin.org/en/developer-reference#getrawtransaction> (הסבר על שדות ההעברה האמיתיים).

החלוקה של הביטים של העברה כלשהי מתחלקת ל- כותרת (2 שדות), קלט (4 שדות) ופלט (2 שדות לכל כתובת).

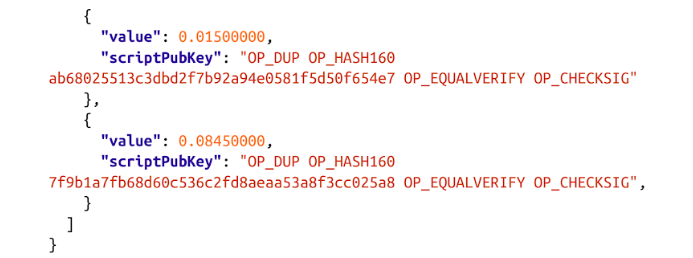
כדי לראות את החלוקה של ההעברה, נוכל לכתוב בbitcoin core:

$ bitcoin-cli getrawtransaction

וכיוון שלא ציינו מספר העברה (הארנק יכול לתת מידע רק עבור העברות שביצע אלא אם נגדיר במפורש אחרת) נקבל פורמט להעברה:



נוכל לראות שהעברה מתחלקת ל3 חלקים: headers לפני ואחרי, מערך קלט (vin), מערך פלט (vout).



על מנת להבין טוב יותר מאיפה הנתונים נשאבים ואת פעולת ה"ארנק", נצטרך להבין את מושג הUTXO – פלט יתרת ההעברה.

הUTXO הוא יחידה של ביטקוין קבועה. לחלק אותה לחלקים מתאפשר רק באמצעות הארנק. מכאן, כששולחים סכום מכתובת מסויימת, אין טעם לציין את יתרת המטבע. יתרת המטבע לקוחה מההעברה הקודמת והיא מההעברה שקדמה לה עד יצירת המטבע.

חלוקה של העברה לפי ביטים והצגה של מספר סיריאלי:

<https://learnmeabitcoin.com/glossary/transaction-data>

פלט:

הפלט מורכב מ3 שדות לכל כתובת שאליה מועבר כסף:

* סכום
* אורך הסקריפט הנועל
* סקריפט הנועל עצמו

קלט:

הקלט מורכב מ4 שדות מכל כתובת שממנה מועבר כסף:

* מצביע להעברה קודמת שממנה ניקח את המטבע UTXO.
* אינדקס הoutput בהעברה הקודמת
* סקריפט המפענח שיאשר שהעברה הקודמת מקושרת
* מפתח ציבורי של השולח

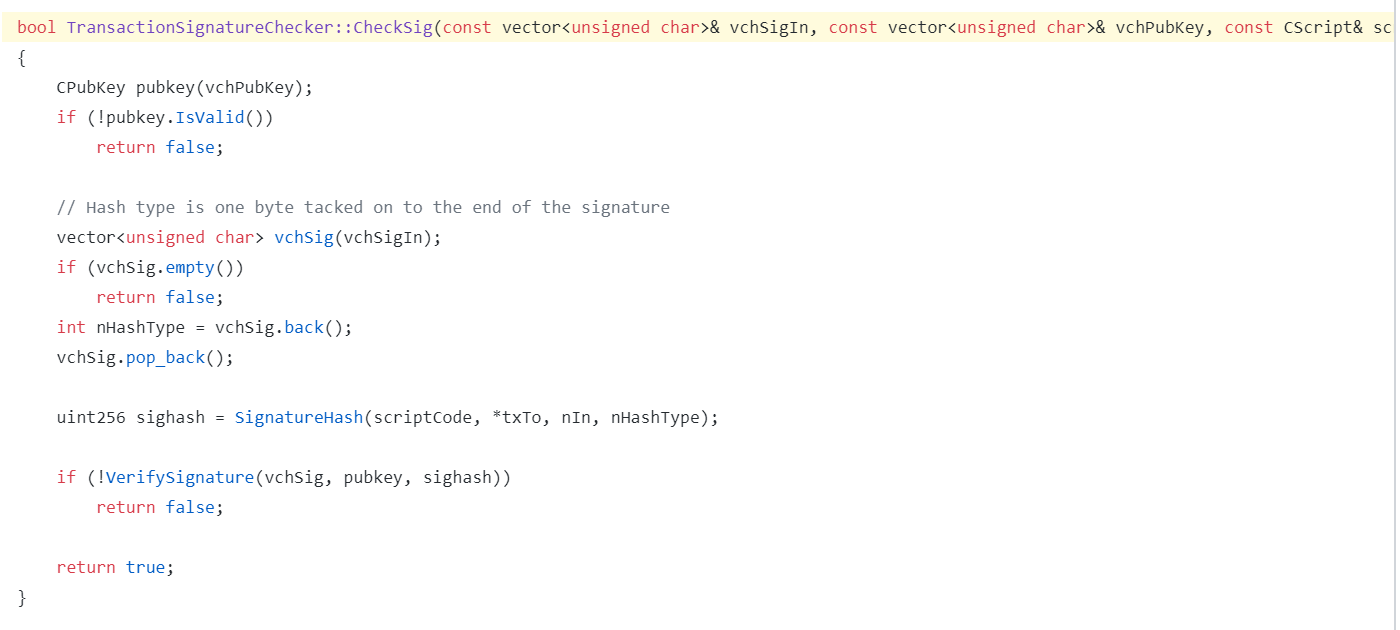
סקריפטים:

מוגדרים כבעלי שלמות טיורינג. בהעברה ישנם 2 סקריפטים – סקריפט הקלט וסקריפט ופלט.

2 3 OP\_ADD 2 OP\_MUL 1 OP\_ADD 11 OP\_EQUAL

הוא דוגמא לסקריפט שבודק את התנאי:

Op\_checksig היא הפונקציה הבאה:



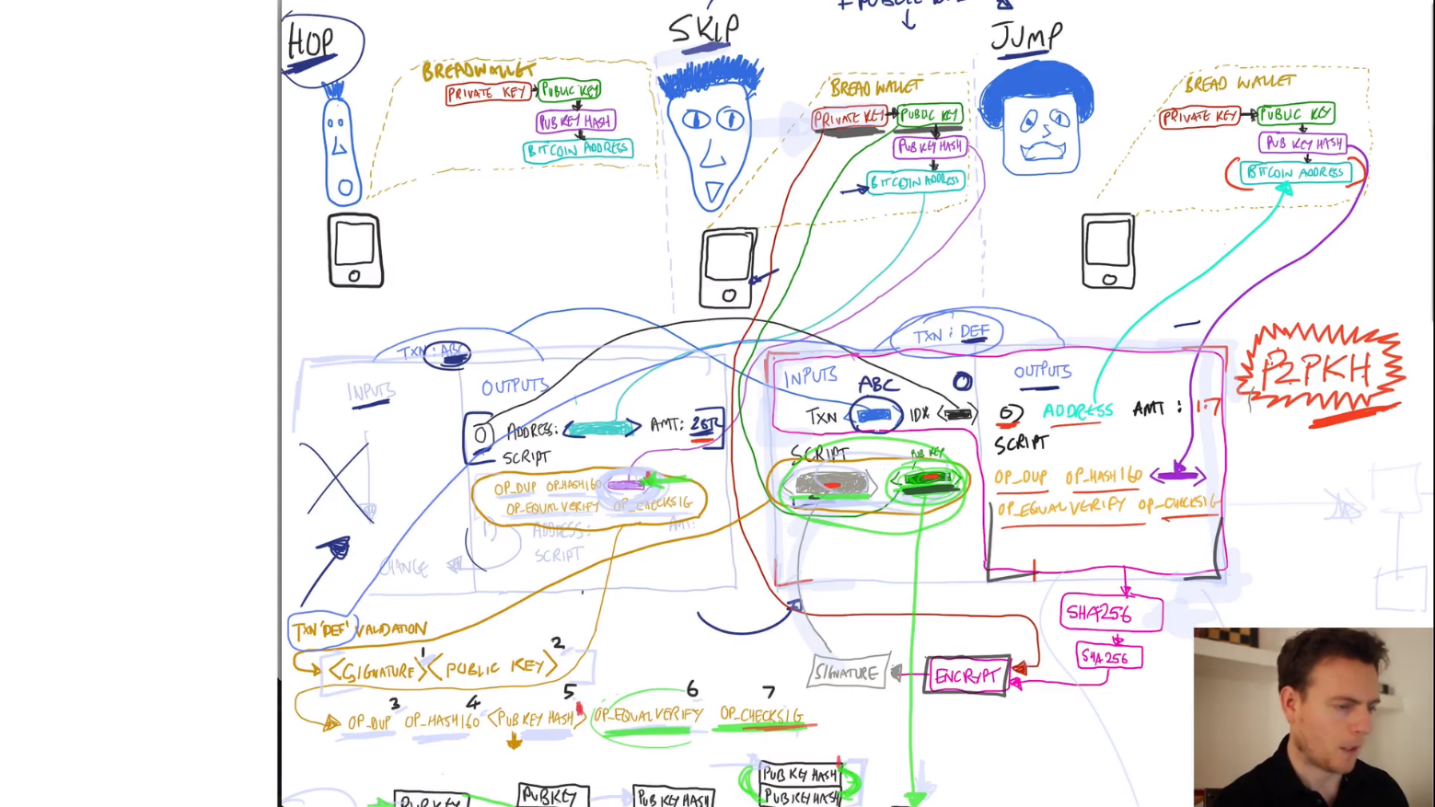
הבודקת את חתימת ההעברה ע"י השולח עם המפתח הציבורי שלו.

החתימה היא תוכן הoutput שמועבר בפעמיים sha256 ומוצפן באמצעות המפתח הפרטי.

הפונקציה בעצם מפענחת את החתימה באמצעות המפתח הציבורי של השולח ובכך מקבלת Hash כלשהו שצריך להיות שווה לפעמיים SHA256 על תוכן הoutput – אם שווה מחזיר אמת, אחרת שקר.

<https://en.bitcoin.it/wiki/Script>

<https://bitcoin.org/en/developer-reference#opcodes>

***חתימה***

פעולת החתימה היא פעולה על עקומה אליפטית כך שבנוסף למפתח הפרטי של השולח, הוא יוצר זוג חדש של מפתחות פרטי וציבורי זמניים עבור כל העברה.

An ECDSA signature is a pair (r,s) where r is the X coordinate of kG, and s = (m+r\*x)/k (where k=nonce, m=message hash, x=private key, G=curve generator)

מלבד הרצת הסקריפט (כאמור, בכל בדיקת העברה הפרוטוקול מריץ את הסקריפט על מנת לוודא אמינות) ישנן עוד **19** הרצות עבור **כל** העברה בכל בלוק, ועוד **19** הרצות עבור כל בלוק בשרשרת הביטקוין.

<https://en.bitcoin.it/wiki/Protocol_rules#cite_note-7>

20 פעולות עבור כל העברה בכל בלוק:

1. בדיקת נכונות תחבירית (עבור מפתח ציבורי)
2. וידוא שקיימת לפחות רשומה אחת ברשימת הקלט וברשימת הפלט (אינן רשימות ריקות)
3. גודל הבלוק קטן מהגודל המקסימלי המוגדר (משתנה)
4. סכום של כל פלט והסכום הכללי הם בגבולות הסכומים האפשריים (legal money range)
5. אף קלט אינו עם הגיבוב 0 (למעט מטבע הבסיס שניתן כפרס על כרייה)
6. nLockTime <= INT\_MAX (זהו שדה חיצוני בheader של פקטת ביטקוין וגם בהעברה יש לפחות 100 בייט והחתימות הכלולות בשדה חתימה בהעברות סטנדרטיות גדול מ2
7. סקריפט אינו סטנדרטי. מבצע פעולות מורכבות מלבד להוציא מהמחסנית ולהכניס אליה וכו'
8. בודק שאין העברה דומה ב"בריכת ההעברות" או בשרשרת הבלוקצ'יין
9. אם ההעברה הקודמת בוזבזה כבר בהוצאה אחרת
10. אם הקישור להעברה הקודמת לא קיים ההעברה נוספת ל"העברות יתומות"
11. אם המטבע הוא מטבע בסיס (יש מאין) מספר האישורים להעברות שנכללו בבלוק הכרייה הוא לפחות 100
12. עבור כל קלט, בדיקה שהקישור לפלט רלוונטי אכן קיים (כפול עבור 8? עבור 9?)
13. בדיקה שערכי הקלט הם בטווח הסכום האפשרי (כמו 4 רק שכאן נדרש ללכת לקישור העברה האחרונה על מנת להגיע לסכום הקלט)
14. בדיקה שערכי הפלט קטנים מערכי הקלט (לא מבוזבז כסף שלא קיים)
15. ההעברה נדחית אם סכום העמלה קטן מכדי להיכנס לבלוק
16. **וידוא הסקריפט הקיים בהעברה שאכן מביא תוצאה נכונה (True)**
17. הוסף את ההעברה ל"בריכת ההעברות"
18. הוסף את ההעברה לארנק כהעברה שנכרית
19. קשר את ההעברה לרשת הביטקוין (peer)
20. עבור כל עסקה יתומה, הרץ את כל השלבים מחדש באופן רקורסיבי